33 Adult Computer Games in BASIC

for the

IBM PC®, APPLE II®/IIe® and TRS-80™

by David W. Chance

33 ADULT COMPUTER GAMES IN BASIC

FOR THE

IBM PC®
APPLE II®/IIe® &
TRS-80™

33 ADULT COMPUTER GAMES IN BASIC

FOR THE IBM PC® APPLE II®/IIe® & TRS-80™

BY DAVID W. CHANCE

TAB BOOKS Inc.
BLUE RIDGE SUMMIT, PA 17214

NOTICES

Apple II, Apple II Plus and Apple //e are registered trademarks of Apple Computer Inc. with regard to any computer product
IBM Personal is a registered trademark of International Business Machines Corporation.
TRS-80 is a trademark of the Radio Shack division of Tandy Corporation.

Program listings for the IBM PC and Apple II systems are provided by Tan Summers.

FIRST EDITION

FIRST PRINTING

Copyright © 1983 by TAB BOOKS Inc.

Printed in the United States of America

Reproduction or publication of the content in any manner, without express permission of the publisher, is prohibited. No liability is assumed with respect to the use of the information herein.

Library of Congress Cataloging in Publication Data

Chance, David.
33 adult computer games in BASIC for the IBM PC®, Apple II/ //e® and TRS-80™
Includes index.
1. Games—Data processing. 2. TRS-80 (Computer)—Programming. 3. Basic (Computer program language)
I. Title. II. Title: Thirty three adult computer games in BASIC.
GV1469.2.C448 1983 794.8'2 83-4881
ISBN 0-8306-0627-0
ISBN 0-8306-1627-6 (pbk.)

Contents

Introduction viii

1 **The Sound Routine** 1
The Sound Routine for the TRS-80—Sound Routine for the Apple II

2 **Games of Skill and Logic** 11
+The Course—Words-Zit--Math-A-Code—*Master Code—Driver's Test—What Is It?

3 **Games of General Interest** 59
Where's The Necessary Room?—Troubles—++General Store—Respect Your Computer—Love One Another—Date Service—Key A Date

4 **Games That Span the Ages** 139
Squares—++*Hold Time—*Dangerous Affair—Pennies or Dollars—++* Be Prepared (Part 1)—++* Rain Of Terror—++Knights

5 **Games to Keep You Alert or Put You to Sleep** 203
*Triangle—++Mind Invasion—++*Trapped—++Elevator—++Farewell—++Insomnia—++Rainy Daze

6 **Games for Parties or Friendly Get-Togethers** 269
Amount On Board—++Lap-The-Track—Partners—+Point A to Point B—++*Injection—+*Intercept

Appendix A Program Listings for the IBM PC 325
Hold Time—Elevator—Farewell—Mind Invasion—Knights—Injection—Point A to Point B—Intercept—Insomnia—Lap-the-Track—Be Prepared—The Course—Trapped—Rainy Daze—General Store—Rain of Terror

Appendix B Program Listings for the Apple II 403
General Store—Hold Time—Be Prepared—Rain of Terror—Knights—Mind Invasion—Trapped—Elevator—Farewell—Insomnia—Rainy Daze—Lap-the-Track—Injection

Appendix C Making the Programs Work on Your Own Computer 469
Screen Formatting—Screen Graphics—Sound Effects—Variable Length

Appendix D Screen Locations 475
TRS-80—Apple II—IBM PC

Appendix E Setting Up Your Games Disk 481
Making A Disk for the Apple II

Index 489

* Program uses the sound routine.
++ Separate listings for the IBM PC and Apple II are included in Appendices A and B
\+ Separate listing for the IBM PC is included in Appendix A.

I would like to dedicate this book to my parents:
James and Imogene Chance

Introduction

Why for adults only? Why not! It seems as though the electronic games of today are geared toward the young. There are not too many adults that will go to the corner arcade to play the latest video craze, so why not have some games that adults can relate to? Here they are! A collection of 33 programs, just for us adults!

Each program marked with an asterisk in the Table of Contents contains a sound output routine (see Chapter 1). This sound output will never compare with that of arcade games, but it keeps your computer from being dead silent! This routine was written on and for a Radio Shack TRS-80 Model I Level II computer system. With a little modification it will work on the Radio Shack TRS-80 Model III Level II computer system. Apple users find a separate sound routine and accompanying explanation at the end of chapter one.

If you have an IBM PC, you can skip the first chapter of this book entirely—the IBM is a born musician. IBM users will find sound (and honest-to-goodness music!) in programs besides those noted in the table of contents. Since the IBM PC doesn't require a machine language routine to create sound and music, adding them is easy. You may find yourself composing background music for every program in the book!

All thirty-three programs in this book were written on and for the TRS-80 computers mentioned above, but notes on converting the programs for the IBM PC and APPLE II+ or *II*e have been

included in Appendix C. Separate listings for the Apple and the IBM have been included for any programs which make extensive use of sound and/or graphics, as noted in the table of contents. The IBM programs were written with the color/graphics adapter in mind, but suggestions are included in Appendix C for using the programs with a monochrome screen.

If you own a computer system different from those mentioned above, you can still enjoy this book. You should be familiar enough with BASIC to know which commands your computer uses, so that you can make any needed changes. Look at the different versions of the more complicated programs. Chances are, one of them will work on your computer with just a few changes. Graphics are the hardest, but since many of the programs use text screen characters, you should be able to adapt them. Look at the charts in Appendix D and visualize the graphics as described in the program; then adapt them to the commands your computer uses.

Most of the programs require at least 16K (kilobytes) of random access memory, but some of the simpler ones use less.

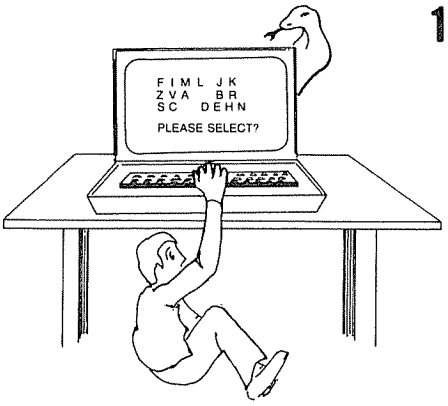
All programs were designed with care, and all have been completely debugged. With each program you'll find:

A brief outline describing in short what the program is about;
A flowchart to help you understand the program;
A sample run to let you see what the program is actually going to do before you run it;
The program listing with ample REM statements in case you decide on making your own changes;
A list of variables and strings and what each perform within the program;
Line explanations indicating line-by-line what the program is doing and what steps each line is performing.

Most of the programs can handle several different players. Some allow up to 10 people to play at once. One program, in the final chapter can handle 15 different players at once!

The final chapter of the book is devoted to parties or friendly get-togethers. Here you'll find six fast action games. If you decide to bet real money, you can be assured that the computer will keep track of each cent. All games within this chapter are fair, meaning that the computer will do no cheating, no one sided favors to keep the same person winning and no printing of false amounts. Of course, all of the programs in the book are like that, except for one.

So sit back, relax, and enjoy! You do not have to hide this book from your children. It contains no adult situations or foul language—just lots of action!



The Sound Routine

The sound routines in this chapter are used to enhance your enjoyment of a number of the game programs in the book. Feel free to use the techniques you learn here to introduce sound into your own programs.

THE SOUND ROUTINE FOR THE TRS-80

There are two ways you can use the sound routine described in this section. You could make a small driver program and place it on a tape different from your program tapes. You could also make a driver program by creating a machine language tape, using the TBUG program or the EDITOR/ASSEMBLER program available at your local Radio Shack store.

If you plan to use sound routines with your TRS-80, you'll need a small amplifier/speaker. Radio Shack's stock number 277-1008 will meet your requirements. If you use TBUG to make your own driver, use the filename SOUND as a title. When you enter it using the system command and the computer returns with ? press the break key and your driver will be ready. (Only those who own a TBUG program need concern themselves with these instructions.)

The method I will discuss now requires nothing more than a blank tape. I will present separate routines for 16K, 32K, and 48K systems. (I will give the routine for 4K at the end of this chapter.) Each routine is a short program in itself and as stated, should be kept on a separate tape.

```
16K -- ANSWER MEM SIZE? 31231

10 FOR I=31232 TO 31268

20 READ A:POKE I,A

30 NEXT

40 POKE 16526,0:POKE 16527,122

50 DATA 221,33,33,122,221,70,0,0,1

60 DATA 37,122,221,70,1,62,1,211,255

70 DATA 16,254,221,70,1,62,2,211,255

80 DATA 16,254,13,194,11,122,30,10,0,201

32K -- ANSWER MEM SIZE? 47615

10 FOR I=47616 TO 47652

20 READ A:POKE I,A

30 NEXT

40 POKE 16526,0:POKE 16527,186

50 DATA 221,33,33,186,221,70,0,0,1

60 DATA 37,186,221,70,1,62,1,211,255

70 DATA 16,254,221,70,1,62,2,211,255

80 DATA 16,254,13,194,11,186,30,10,0,201

48K -- ANSWER MEM SIZE? 63999

10 FOR I=64000 TO 64036

20 READ A:POKE I,A

30 NEXT

40 POKE 16526,0:POKE 16527,250

50 DATA 221,33,33,250,221,70,0,0,1
```

```
60 DATA 37,250,221,70,1,62,1,211,255

70 DATA 16,254,221,70,1,62,2,211,255

80 DATA 16,254,13,194,11,250,30,10,0,201
```

Remember, use only the program that fits your system's memory size. Be sure also that line 40 is used (initialization). This line is very important. You will have no sound output without it.

Each of the programs that contain sound are written for a 16K system. If you have a 32K or 48K system, you will also have to change the variables for sound output. Turn to a program that contains sound output. You will find that the variable (for instance MC in line 370 in the Master Code program, or SS in line 295 in the Hold Time program) is set to 31266. For a 32K system, change the figure (and all others in the other programs) to 47650. For a 48K system change that figure to 64034. I designed the programs for a 16K machine because I believe this is the size system most people buy. (I could be wrong!) If I had included figures for the other systems, it would have made the programs confusing. Just be sure to change these figures for a 32K or 48K and you'll encounter no problems.

If you've created the sound program for your system, you are now ready to try some experiments with sound. Remove the plug that is marked AUX from your recorder and plug it into the small amplifier/speaker that I described earlier. (Be sure there is a fresh battery in it.) Turn it on to a moderate to low volume.

```
10 REM EXPERIMENTS WITH SOUND

20 N=10:A=31266:REM A=47650 FOR A 32K - A=64034 FOR A 48K

30 POKE A,N

40 X=USR(0):REM CALL ROUTINE

50 FOR T=1 TO 30:NEXT:REM PAUSE

60 IF N>=255 THEN 80:REM 255 LIMIT FOR VARIABLE

70 N=N+1:GOTO 30

80 END
```

Now run the program. You should hear sounds coming from the amplifier/speaker. Adjust the volume on it if necessary. If you do not have sound, recheck the driver (machine program) to see if you entered all figures correctly and make sure that the initialization (line 40) has been completed. If you are receiving sound, the pitch will continue to become lower as the variable N becomes higher. When N reaches 255, the program will end. Never use a number for N that is below 10 or greater than 255.

For a dual output, add these lines to the program in residence:

```
41  IF N<= 254 THEN POKE A,N+1 : X = USR(0)
```

Run the program with the above line, then add:

```
42  POKE A,N-1 : X = USR(0)
```

This time delete the pause loop, line 50, and run again. Any time you want to stop the program (if N<255) press the break key. For strange sounds, add these two lines and run the program.

```
43  Q = RND(255):IF Q<10 THEN 43
44  POKE A,Q : X = USR(0)
```

After experimenting with the above, delete lines 41-80 and add the following:

```
50 A$=INKEY$

60 IF A$="" THEN 30

70 IF N>=255 THEN 90

80 N=N+1:GOTO 30

90 END
```

The complete program should look like this:

```
10 REM EXPERIMENTS WITH SOUND

20 N=10:A=31266:REM A=47650 FOR 32K - A=64034 FOR 48K

30 POKE A,N
```

```
40 X=USR(0):REM CALL ROUTINE

50 A$=INKEY$

60 IF A$="" THEN 30

70 IF N>=255 THEN 90

80 N=N+1:GOTO 30

90 END
```

Now run the program above. Each time you press any key, the tone gets lower. After you've experimented with this program, type NEW and enter the following program:

```
10 REM MAKE A TUNE

20 DIM N(100):REM FOR UP TO 100 NOTES

30 REM WILL USE KEYS A,B,C,D,E,F,G

40 A=10:B=50:C=80:D=100:E=150:F=200:G=225

50 I=1:AA=31266:REM A=47650 FOR 32K - A=64034 FOR 48K

60 CLS:PRINT"TO HEAR YOUR TUNE"

70 PRINT"PRESS THE SPACE BAR."

80 PRINT"START COMPOSING, USE THE"

90 PRINT"LETTERS A,B,C,D,E,F,G"

100 A$=INKEY$

110 IF A$="" THEN 100

120 IF A$=" " THEN 310:REM SPACE BAR

130 IF A$="A" THEN 210

140 IF A$="B" THEN 220

150 IF A$="C" THEN 230

160 IF A$="D" THEN 240

170 IF A$="E" THEN 250
```

```
180 IF A$="F" THEN 260

190 IF A$="G" THEN 270

200 GOTO 100

210 POKE AA,A:N(I)=A:GOTO 280

220 POKE AA,B:N(I)=B:GOTO 280

230 POKE AA,C:N(I)=C:GOTO 280

240 POKE AA,D:N(I)=D:GOTO 280

250 POKE AA,E:N(I)=E:GOTO 280

260 POKE AA,F:N(I)=F:GOTO 280

270 POKE AA,G:N(I)=G

280 X=USR(0):REM CALL ROUTINE

290 IF I>=100 THEN 310:REM N(I) FOR NOTES IN TUNE

300 PRINT A$;:I=I+1:GOTO 100

310 T=I-1:PRINT

320 PRINT"SPEED OF PLAYBACK (FAST,MEDIUM,SLOW)"

330 INPUT"1,2 OR 3";TT:TT=TT*20

340 FOR I=1 TO T:IF N(I)=0 THEN 370

350 POKE AA,N(I):X=USR(0)

360 FOR P=1 TO TT:NEXT P,I

370 PRINT

380 PRINT"END OF YOUR TUNE."

390 END
```

Run the above program and compose different tunes. You may be able to tell from the tones that I am no musical expert! You can experiment with the constants at line 40, A-G, to come up with some tones of your own. Remember to keep these constants between 10 and 255, 10 will be the highest tone you can achieve, and 255 will be

the lowest. Of course, you have all the numerals between 10 and 255 to experiment with. The program lines are explained below.

Line 20 dimensions the N array so that 100 notes can be placed into your tune. You can either increase or decrease this amount, depending on what you are planning to compose.

Line 30 is a remark indicating which keys will be used.

Line 40 defines the constants for the 7 different notes (A-G). You can change these to whatever you choose as long as the numbers are between 10 and 255.

Line 50 sets or initializes the counter I and the poke location, AA. Always remember that these constants are geared to a 16K system, here, and throughout the book. If you have a 32K system, this constant will be 47650. If your system is a 48K system, the constant will be 64034.

Lines 60-90 are simple print statements.

Line 100 sets up the INKEY$ function for the keys that you will press.

Line 110 causes the program to loop to line 100 until a key is pressed.

Lines 120-190 identify the key that was pressed. Whenever you press the space bar, the computer assumes that all of your notes are in and the tune is replayed for you. Otherwise, the computer will only notice keys that are within the arguments and branch to the line specified when they are pressed.

Line 200 loops back to line 100. This assures that the program will not fall, continue if any key other than those specified is pressed.

Lines 210-270 poke the value of the constant for the key that was pressed into location AA. This value will then be placed into the subscripted variable N(I) for playback. The tune you enter will be played back exactly as you input it.

Line 280 will call the sound routine so you'll hear the note (sound) that you have selected immediately. The letter X can be any other letter that you choose so long as it is not used anywhere else within a program; that is, if you want you could use the letter Z:Z=USR(0) would be valid for the above program, but if you use A = USR(0), you would lose the constant you had for A and an error would occur in your program.

Line 290 will test the amount of counter I. If it has reached 100, the limit for the number of notes, the program will branch to line 310.

Line 300 will print the key you have pressed (if valid), advance

the counter by 1, and return to line 100 for another key closure.

Line 310 initializes the variable I to I-1 for the playback of all the notes you entered, provided that the space bar was pressed in line 120.

Lines 320-330 ask you at what speed you want your tune replayed. The amount you enter, 1, 2, or 3, will be multiplied by 20 and used in a for-next loop throughout the playback. You can experiment with different values of TT if you so choose. For example, TT=TT*20 could be changed to TT=TT*10, for a faster playback.

Lines 340-360 contain 2 loops: one for notes entered and the other for a pause during playback. The second statement in line 340 will end the program if one of the notes is equal to 0 (zero). It's unlikely that one would be, but the statement is there to be on the safe side. Line 350 then pokes the value that is in the appropriate location in the N(I) array into location AA and calls the sound routine with X=USR(0). Line 360 forms the pause or delay loop used during playback, and also contains the next statement for the notes (I).

Lines 370-390 terminate the program when all notes have been played back.

From all these examples, you should have the general idea of how to use your sound routine. If not, you should review the material before you tackle writing programs that will include sound. If you plan to write your own programs with sound or to add sound to some that you now have, do not add the sound routine to a program that uses the CHR$(23) function (double size). Take my word for it, it will not work; you will receive nothing more than error messages.

If you own a 4K system and you're still scratching your head wondering what the sounds are like, you're in luck! Here is the driver for the 4K system (LEVEL II). Remember, the programs in this book are intended for machines with at least 16K of RAM. This 4K driver doesn't mean you can enter and run the other programs.

```
ANSWER MEM SIZE? 20223

10 FOR I=20224 TO 20260

20 READ A:POKE I,A

30 NEXT

40 POKE 16526,0:POKE 16527,79

50 DATA 221,33,33,79,221,70,0,0,1
```

```
60 DATA 37,79,221,70,1,62,1,211,255

70 DATA 16,254,221,70,1,62,2,211,255

80 DATA 16,254,13,194,11,79,30,10,0,201

To receive sound POKE a numeral between 10 and 255 into

location 20258 and CALL the routine. Example:

100 A=20258

110 POKE A,50:X=USR(0)
```

Enter and run all of the above. You will hear a short beep from the amplifier/speaker. Remember that the address that you will poke a numeral into is 20258. Anytime you desire to hear a tone, use USR(0) (CALL,X=USR(0)). You can enter and run all of the short programs that are in this chapter on the 4K system.

For all of the systems, remember to keep the driver that fits your machine RAM on a separate tape to use whenever you are both to use a sound output on a program and always answer MEM SIZE? with the correct constant when booting-up your system to protect the area where the sound routine is residing. Once loaded into memory, the driver will remain there until the power is turned off. It cannot be lost when you type NEW or enter a different program through BASIC.

To use any of the programs that contain sound without the sound output, delete all of the U=USR(0) statements (machine language calls) from the program.

SOUND ROUTINE FOR THE APPLE II

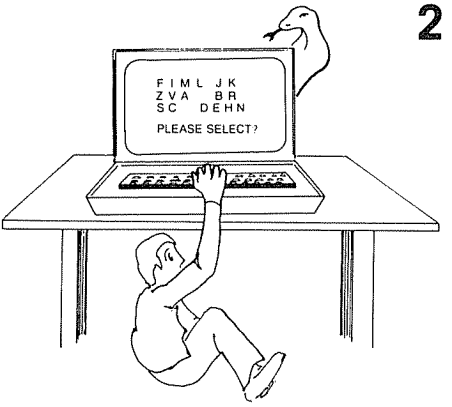
The following program for the Apple II computer was written and tested on a 48K system. To use it, you may either type it in at the beginning of each program using sound—a lot of work, but simpler at "run-time," or save it on a disk to be loaded each time it is needed for a program. The Apple II versions of programs in this book contain the commands to load and "call" the routine with the assumption that you will have it saved on the same disk with your programs.

To use the routine on your program disk, first type in the program below and run it. The program will take less than a minute to run. then type in the following command: BSAVE SOUND,

A$300,L$1E and press the return key. The program will now be on your program disk, ready to use.

```
SOUND

10  FOR I=768 TO 797

20    READ A:POKE I,A

30  NEXT

40  DATA 0,0,0,172,1,3

50  DATA 174,1,3,173,2,3

60  DATA 32,168,252,173,48,192

70  DATA 232,208,253,136,208,238

80  DATA 206,0,3,208,230,96
```

The SOUND program for the Apple II is based on material developed by Jesse Sturgis and is used by permission.



Games of Skill and Logic

The Course challenges you to react quickly in order to maneuver your vehicle safely down an ever-changing course.

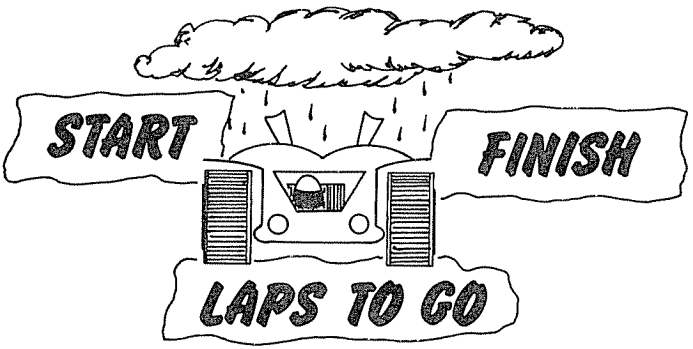
Words-Zit presents coded messages for you to decipher. The code is new each time you load the program.

Math-a-Code gives you a chance to test your prowess in addition . . . but each digit is represented by a letter!

Master Code is another code game, but this time each letter is represented by a different tone.

Driver's Test allows you to test your reflexes and avoid collisions without risking the family car.

What Is It? is a competitive game that tests your ability to guess words using clues that become progressively more specific.



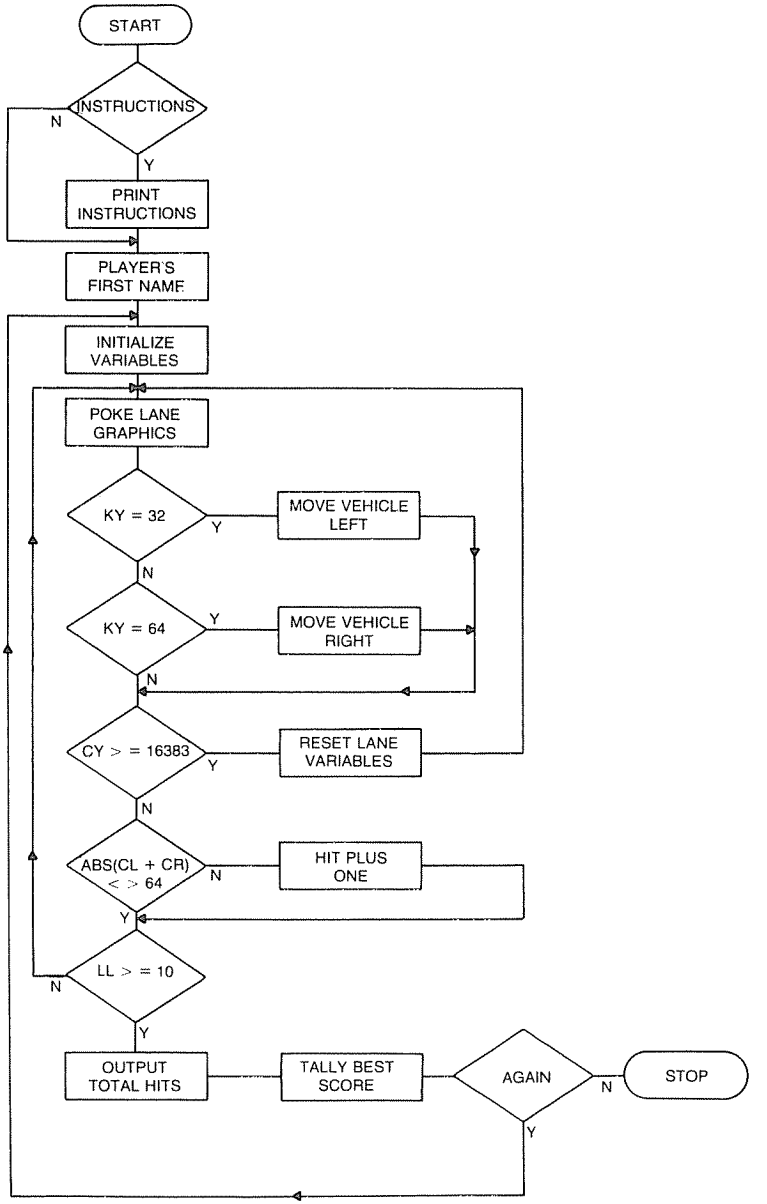
THE COURSE

You will be required to have top-notch driving abilities as you negotiate the 10 laps of this challenging course. Be careful not to make the slightest contact with the edge of the track because each time you do, it will count as a hit. You will control the vehicle with the left and right arrow keys. Note that the IBM version of this program is included in Appendix A.

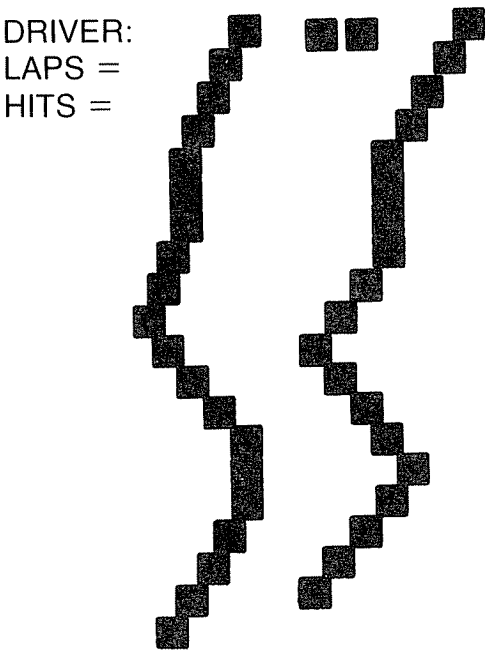
Sample Run

```
>>> THE COURSE <<<

REQUIRE INSTRUCTIONS? YES

THE COURSE WILL REQUIRE THAT YOU
USE TOP NOTCH REFLEXES TO CONTROL
YOUR VEHICLE THROUGH A DISTANCE OF
10 LAPS, THE COURSE WILL BE BOTH
STRAIGHT AND WINDING, YOU WILL
CONTROL YOUR VEHICLE WITH THE
LEFT AND RIGHT ARROW KEYS.
THE SLIGHTEST CONTACT WITH THE
EDGE OF THE TRACK WILL CAUSE
AN ACCIDENT, AND A HIT WILL BE
ADDED TO YOUR SCORE.
```

PRESS A KEY . . .
FIRST NAME OF DRIVER? WALLACE
WALLACE STRIKE A KEY TO BEGIN . . .



Flowchart for *The Course*.



Program Listing

```
 10 REM PROGRAM TITLE: THE COURSE
 20 CLS:PRINT TAB(10);
 30 PRINT")))  THE COURSE  ((("
 40 PRINT
 50 INPUT"REQUIRE INSTRUCTIONS";X$
 60 IF X$="N" OR X$="NO" THEN 210
 70 PRINT
 80 PRINT"THE COURSE WILL REQUIRE THAT YOU"
 90 PRINT"USE TOP NOTCH REFLEXES TO CONTROL"
100 PRINT"YOUR VEHICLE THROUGH A DISTANCE OF"
110 PRINT"10 LAPS.  THE COURSE WILL BE BOTH"
120 PRINT"STRAIGHT AND WINDING.  YOU WILL"
130 PRINT"CONTROL YOUR VEHICLE WITH THE"
140 PRINT"LEFT AND RIGHT ARROW KEYS."
150 PRINT"THE SLIGHTEST CONTACT WITH THE"
160 PRINT"EDGE OF THE TRACK WILL CAUSE"
170 PRINT"AN ACCIDENT, AND A HIT WILL BE"
180 PRINT"ADDED TO YOUR SCORE."
190 PRINT:PRINT:PRINT"PRESS A KEY..."
```

```
200 X$=INKEY$:IF X$="" THEN 200
210 CLS:T=1:AT=1
220 INPUT"FIRST NAME OF DRIVER";D$
225 IF D$="" THEN PRINT:GOTO 220
230 PRINT D$;" STRIKE A KEY TO BEGIN..."
240 X$=INKEY$:IF X$="" THEN 240
245 IF T)=2 THEN CLS:GOTO 260
250 CLS:DIM CX(18),CY(18)
260 PRINT@0,"DRIVER: ";D$;:GOSUB 1000
265 CC=0:LP=0:LL=0:HI=0:' THE COURSE
270 CX=15388:CY=CX+9:J=191:C=1
275 C1=CX+3:C2=C1+2:' VEHICLE
280 IF CY)=16383 OR C1)=16383 THEN 400
285 IF CC=1 THEN 420
290 POKE CX,J:POKE CY,J
295 CX(C)=CX:CY(C)=CY:' TO BLANK OLD COURSE
300 CX=CX+64:CY=CY+64
305 C=C+1
310 DD=RND(2)
315 KY=PEEK(14400):' KEYS
320 IF DD=1 THEN 340
330 CX=CX+DD-1:CY=CY+DD-1:GOTO 360
340 DD=-DD
350 CX=CX+DD:CY=CY+DD
360 ' CHECK FOR MOVEMENT (VEHICLE)
370 IF KY=32 THEN 500
380 IF KY=64 THEN 530
390 GOTO 280
400 CC=1:LP=LP+1
405 IF (LP/10)=.6 THEN LL=LL+1:LP=1
410 PRINT@64,"LAPS =";LL;
415 IF LL)=10 THEN 700 ELSE GOTO 270
420 POKE CX(C),32:POKE CY(C),32
430 POKE C1,32:POKE C2,32
440 IF C2)=16319 THEN 290
450 C1=C1+64:C2=C2+64
460 POKE C1,183:POKE C2,187
470 CL=PEEK(C1-1):CR=PEEK(C2+1)
480 IF ABS(CL+CR)<>64 THEN 600
485 GOTO 290
500 ' VEHICLE MOVEMENT (LEFT)
505 POKE C1,32:POKE C2,32
510 C1=C1-1:C2=C2-1
520 GOTO 280
530 ' VEHICLE MOVEMENT (RIGHT)
535 POKE C1,32:POKE C2,32
540 IF C2)=16319 THEN 280
550 C1=C1+1:C2=C2+1
```

```
 560 GOTO 280
 600 ' HITS TO VEHICLE
 610 HI=HI+1
 620 PRINT@128,"HITS =";HI;
 630 GOTO 290
 700 ' FINISH
 710 PRINT@192,CHR$(30)
 720 IF HI=0 THEN 800
 730 PRINT"YOU DIDN'T MAKE THE";LL;"LAPS"
 740 PRINT"WITHOUT ERROR. YOU TOOK A"
 750 PRINT"TOTAL OF";HI;"HITS...."
 760 PRINT"TRY AGAIN, ";D$;
 770 INPUT X$
 780 IF X$="N" OR X$="NO" THEN 1100
 790 AT=AT+1:PRINT:GOTO 830
 800 PRINT"NOT BAD! NO HITS AT ALL!"
 810 PRINT"THINK YOU MIGHT WANT TO"
 820 GOTO 760
 830 T(T)=HI:' BEST SCORE
 835 IF T<=1 THEN 920
 840 I=1:Z=0
 850 IF T(I)<=T(I+1) THEN 880
 860 YS=T(I):T(I)=T(I+1):T(I+1)=YS
 870 Z=1
 880 I=I+1:IF I>=T THEN 900
 890 GOTO 850
 900 IF Z=1 THEN 840
 910 BS=T(1)
 920 T=T+1:IF T>=5 THEN BS=T(1):T=3
 930 PRINT
 940 PRINT:GOTO 230
1000 ' MESSAGE
1010 IF T<=2 THEN 1040
1020 PRINT@42,"BEST SCORE: ";BS;
1030 PRINT@106,"ATTEMPTS: ";AT;
1040 RETURN
1100 ' END
1110 CLS
1120 PRINT"END OF PROGRAM RUN..."
1130 END
```

The Variables and Strings

X$ - General purpose for input commands
T - For users' scores
AT - Total user attempts
D$ - Name of driver
CX & CY (subscripted) for left and right sides of the lane

CC - General purpose variable. When CC is equal to 1, the previous track will be reset as a new course is poked.
LP - When this variable is divided by 10, one lap will have been completed
LL - Total laps
HI - Total hits per 10 lap period
CX, CY & J - For lane placement, poked into video memory
C1 & C2 - For shape of vehicle, poked into video memory
C - Used in conjunction with subscripted variables CX & CY to poke previous track with blanks, CHR$(32)
DD - A random amount to move track (shape) left or right
KY - For user key closure: Left arrow to move vehicle left and right arrow to move vehicle right
CL & CR - Used to detect when a user's vehicle has hit the side of the lane
T(I) - For users' best run
Z - General purpose, used to insure that the best run is correct
BS - Best score obtained

Explanation of the Program Lines

Lines 20-200 print all instructions, if they are needed.

Line 210 initializes two of the variables.

Lines 220-240 asks for the user's first name and a key press to begin the game.

Line 245 skips the dimension statements, if they have already been used.

Line 250 is the dimension statement for the subscripted variables CX and CY, which poked the locations of the left and right sides of the lane.

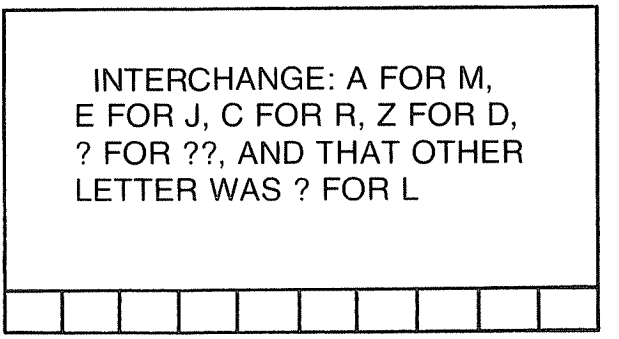
Line 260 prints the driver's name and then branches to 1000 (GOSUB) where best score and total attempts are printed if the variable T is greater than or equal to 2.

Line 265 initializes more variables: CC for the blanking of the old or previous lane, LP for each lap, LL for total laps, and HI for total hits.

Lines 270-630 poke the left and right sides of the lane until the bottom of the video is reached. Then the vehicle is poked at the top of the video. The old course is then followed, using the left and right arrow keys, and the vehicle moves toward the bottom of the video. While all of this is taking place, the old lane (behind the vehicle) is blanked out and a new course is poked. The user will not be grounded for 'hits' to the old lane, just the new, which is being poked

while he or she is driving. This is decided by the absolute amount of CL+CR. If this amount is not equal to 64, a hit has been made to either side of the lane, If LP/10=6, one lap has been completed, and LL is then advanced by one. If LL>=10 (laps) the program branches to line 700, where the score is tallied and the user is asked if he or she wants to try again.

Lines 840-900 determine the best score. The program continues to loop here until T(1) is the lowest number of hits. When equal to 1 variable Z indicates that the lowest has not been reached, so the program branches back to line 840 and begins again.

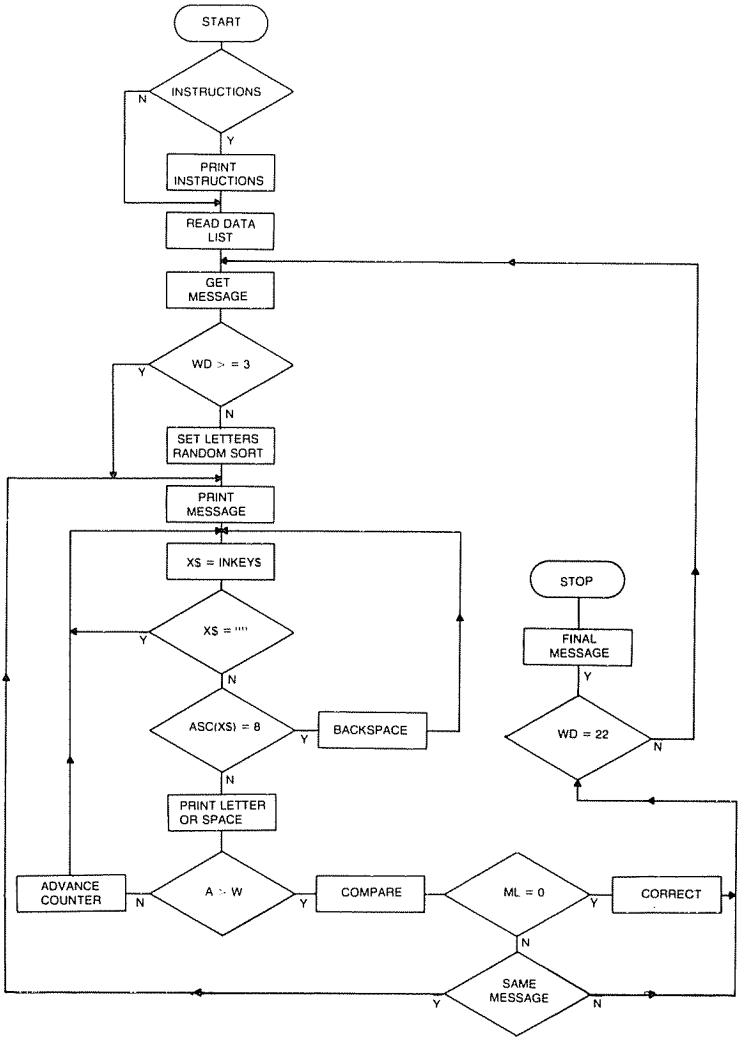


WORDS-ZIT

In this game you'll be faced with decoding 20 different messages formed by interchanging all of the letters of the alphabet, except for E, I and O. Each message will look like a bunch of garbage, but after you have decoded the first message, the last 19 shouldn't be so hard! The program uses INKEY$. If you make a mistake, use the left arrow key to backspace over your mistake and then continue.

Sample Run

```
INSTRUCTIONS? YES
WORDS-ZIT
YOU WILL BE FACED WITH 20 DIFFERENT
MESSAGES TO DECODE. ALL THE
LETTERS OF THE ALPHABET HAVE
BEEN INTERCHANGED, EXCEPT FOR E,
I, AND O, WHICH REMAIN THE SAME.
IF YOU MAKE IT THROUGH THE FIRST
MESSAGE, REMEMBER WHICH LETTERS
WERE CHANGED WITH WHICH, THEN THE
FOLLOWING 19 SHOULDN'T BE SO HARD.
TYPE SLOWLY, DON'T BE IN A HURRY.
THE PROGRAM USES INKEY$, IF YOU
MAKE A MISTAKE, USE THE LEFT
ARROW KEY TO BACKSPACE.
PRESS ENTER?

GOH MLFE DMIZ ONE SIAMD
```



Flowchart for *Words-Zit*.

THE MESSAGE?
I DO NOT HAVE ANY IDEA

YOU HAVE MISSED 19 LETTERS.
TRY THE SAME ONE OR GO TO THE NEXT
ENTER SAME OR NEXT? NEXT

IR DMEG QESE LBB LZ ZIKPBE

THE MESSAGE?
IS MEG IN THE HOUSE NOW

YOU HAVE MISSED 23 LETTERS.
TRY THE SAME ONE OR GO TO THE NEXT
ENTER SAME OR NEXT? NEXT

ULN GOH NOD QOSJ RLZDES

THE MESSAGE?
YOU HAVE SEEN THE RIDER

YOU HAVE MISSED 20 LETTERS.
TRY THE SAME ONE OR GO TO THE NEXT
ENTER SAME OR NEXT? NEXT

GOH MLFE KYLE ID DMIZ RLS

THE MESSAGE?
SCRAMBLE A LITTLE MORE
YOU HAVE MISSED 23 LETTERS.
TRY THE SAME ONE OR GO TO THE NEXT
ENTER SAME OR NEXT? NEXT

STOP

(This is not a complete run. Messages were guessed at so as to reveal the code.)

Program Listing

```
10 ' PROGRAM TITLE: WORD-ZIT
20 RANDOM:CLEAR 700
30 CLS:INPUT"INSTRUCTIONS";A$
40 IF A$<>"Y" AND A$<>"YES" THEN 200
50 PRINT"WORDS-ZIT"
60 PRINT"YOU WILL BE FACED WITH 20 DIFFERENT"
70 PRINT"MESSAGES TO DECODE. ALL THE"
80 PRINT"LETTERS OF THE ALPHABET HAVE"
90 PRINT"BEEN INTERCHANGED, EXCEPT FOR E,"
100 PRINT"I, AND O, WHICH REMAIN THE SAME."
110 PRINT"IF YOU MAKE IT THROUGH THE FIRST"
120 PRINT"MESSAGE, REMEMBER WHICH LETTERS"
130 PRINT"WERE CHANGED WITH WHICH, THEN THE"
140 PRINT"FOLLOWING 19 SHOULDN'T BE SO HARD."
150 PRINT"TYPE SLOWLY, DON'T BE IN A HURRY,"
160 PRINT"THE PROGRAM USES INKEY$. IF YOU"
170 PRINT"MAKE A MISTAKE, USE THE <LEFT"
180 PRINT"ARROW> KEY TO BACKSPACE."
190 INPUT"PRESS ENTER";A$
200 CLS
210 DIM U$(21),M(21),X$(27),L$(27)
220 DIM W$(21),L(27):WD=1
230 GOSUB 4000:W$=W$(X):W=LEN(W$):' MESSAGE
240   SET LETTERS THEN REDO
250 A=1:I=65
260 X$(A)=CHR$(I):L(A)=A
270 A=A+1:I=I+1
280 IF I=91 THEN 300
290 GOTO260
300 A=1:PRINT@0,STRING$(14,32)
305 IF WD>=3 THEN 470
310 X=RND(26)
320 IF L(X)=0 THEN 310
330 L$(A)=X$(X):L(X)=0
340 A=A+1
350 IF A=27 THEN 370
360 GOTO 310
370 ' PLACE E,I, & O IN PROPER PLACE
380 Z$="E":M=5
390 FOR I=1 TO 27
400 IF L$(I)=Z$ THEN 420
410 NEXT
420 ' NOW INTERCHANGE EACH
430 L$=L$(M):L$(M)=L$(I):L$(I)=L$
440 IF Z$="E" THEN Z$="I":M=9:GOTO 390
450 IF Z$="I" THEN Z$="O":M=15:GOTO 390
```

```
460 A=1:' PRINT MESSAGE
470 FOR I=1 TO 26
480 IF X$(I)=MID$(W$,A,1) THEN 510
490 IF MID$(W$,A,1)=" " THEN T$=" ":GOTO 520
500 NEXT
510 T$=L$(I)
520 PRINT T$;
530 A=A+1
540 IF A>W THEN 560
550 GOTO 470
560 PRINT:PRINT
570 PRINT"THE MESSAGE ?"
580 P=256:A=1
590 X$=INKEY$
600 PRINT@P,CHR$(143);
610 FOR T=1 TO 50:NEXT
620 PRINT@P," ";
630 FOR T=1 TO 25:NEXT
640 IF X$="" THEN 590
650 IF ASC(X$)=8 THEN 690
660 PRINT@P,X$;:P=P+1:A=A+1:GOTO 750
670 IF A>W THEN 900
680 GOTO 590
690 P=P-1:A=A-1:' BACKSPACE
700 IF P<=256 THEN P=256:A=1
710 ' BACKSPACE TOTAL MESSAGE
720 Q$=LEFT$(Q$,A-1)
730 GOTO 590
750 ' TOTAL MESSAGE
760 Q$=Q$+X$
770 GOTO 670
900 ' ONE COMPLETED
910 PRINT:PRINT
920 IF Q$<>W$ THEN 1000
930 PRINT"YOU HAVE THIS ONE CORRECT."
935 CR=CR+1:IF WD>=3 THEN 960
940 PRINT"DO YOU REMEMBER THE CODE FOR"
950 PRINT"EACH OF THE LETTERS..."
960 FOR T=1 TO 3000:NEXT
965 IF WD=21 THEN 1200
970 CLS:Q$="":GOTO 230
1000 ' INCORRECT LETTERS
1010 FOR I=1 TO LEN(W$)
1020 IF MID$(Q$,I,1)<>MID$(W$,I,1) THEN 1040
1030 NEXT:GOTO 1050
1040 ML=ML+1:GOTO 1030
1050 PRINT"YOU HAVE MISSED";ML;"LETTERS."
1060 PRINT"TRY THE SAME ONE OR GO TO THE NEXT"
```

```
1070 INPUT"ENTER SAME OR NEXT";A$:ML=0
1080 IF A$="SAME" THEN 1110
1090 IF A$="NEXT" THEN SK=SK+1:GOTO 960
1100 PRINT:GOTO 1060
1110 Q$=""
1120 CLS
1130 PRINT:GOTO 460
1200 ' END OF LIST
1210 CLS
1220 PRINT"SORRY, WE HAVE USED THE ENTIRE"
1230 PRINT"LIST OF MESSAGES..."
1240 PRINT"YOUR FINAL SCORE:"
1250 ' CORRECT / INCORRECT
1260 PRINT CR;"MESSAGES CORRECT."
1270 PRINT SK;"MESSAGES INCORRECT OR SKIPPED."
1280 PRINT
1290 PRINT"END OF RUN."
1300 END
4000 ' READ DATA LIST IF NOT DONE
4010 IF WD>=2 THEN 4030
4020 FOR I=1 TO 20:READ U$(I):M(I)=I:NEXT
4030 ' GET RANDOM MESSAGE
4040 X=RND(20):IF M(X)=0 THEN 4040
4050 W$(X)=U$(X):M(X)=0
4060 ' COUNTER ADVANCE
4070 WD=WD+1
4080 PRINT@0,"PROCESSING..."
4090 FOR T=1 TO 1000:NEXT
4100 RETURN
5000 DATA YOU HAVE THIS ONE RIGHT
5010 DATA YOU MUST BE GETTING BETTER
5020 DATA DOING FINE
5030 DATA A ROLLING STONE DOES NOT GET WET
5040 DATA YOU HAVE MADE IT THIS FAR
5050 DATA YOU MUST BE A CRACK AT CODES
5060 DATA WORDS OR MESSAGES YOU MUST KNOW
     THE CODE
5070 DATA IF THEY WERE ALL AS SIMPLE
5080 DATA IF YOU WANT TO QUIT THINK AGAIN
5090 DATA WE ARE JUST BEGINNING
5100 DATA CAN YOU NOT WORK FASTER
5110 DATA PERHAPS YOU ARE BEGINNING TO
     DEHYDRATE
5120 DATA PERHAPS YOU HAVE TO WORK
     TOMORROW
5130 DATA HIGH ON A MOUNTAIN BREAKING A CODE
5140 DATA IF YOU MADE IT THIS FAR BE A SPY
5150 DATA MICRO CHIPS ALL OVER THE FLOOR
```

```
5160 DATA HAVE YOU THOUGHT OF ENTERING A
     SCHOOL OF CODE
5170 DATA ALL THE TIDES AND A FULL MOON
     COULD NOT STOP YOU NOW
5180 DATA SAY YES IF YOU WANT TO QUIT AND I
     WILL NOT RESPOND
5190 DATA A ZEBRA MUST HAVE STRIPES FOR THE
     FBI FILES
```

Variables and Strings

A$ - General input
U$(I) - For messages
M(I) - For used messages
X$(I) - Correct order of letters
L$(I)-Random order of letters L(I) - Used letters
W - Number of letters and spaces per message (LEN)
A - General counter
I - General purpose, for letters
WD - Counter, messages used
X - Random select/placement of letters into L$(I)
Z$ - For placement of letters E,I and O into their proper locations
T$ - Letters of message (coded)
P - Placement of letters (input) at video PRINT@locations
X$ - For user letter entries
Q$ - User's input message
CR - Counter, correct messages
ML - Counter, missed letters
SK - Messages skipped or incorrect

Explanation of the Program Lines

Line 20 clears 700 bits of memory for string storage (Q$).

Lines 30-200 print the general instruction.

Lines 210-220 dimension the variables and strings and initialize the counter for the messages used.

Line 230 contains a GOSUB to line 4000 where the data list is read into the U$ array if this hasn't already been done. A random message is selected and then deleted from the list so it will not be reselected. The return is made to line 230 where W$ is set to the contents of the W$(X) message, and W is set to the length of W$.

Lines 240-290 sets the subscripted string, X$, to the 26 letters of the alphabet.

Lines 300-360 select random letters from the alphabet, and places them within L$(A).

Lines 370-450 check the location of the letters E, I, and O and interchange them with the three correct letters if necessary.

Lines 460-550 print the entire message in coded form.

Lines 560-640 use the INKEY$ function for user entry of the letters. If X$ is not equal to null, the program branches to line 650.

Line 650 tests the ASC value of X$ for a possible backspace entry.

Line 660 prints the contents of X$ at location P. Location P is then advanced by one, and the count of the letters is increased by one (A).

Line 670 checks the letter counter, A. If the value is greater than W, the entire message has been entered.

Line 680 recycles back to INKEY$, if the message is not finished.

Lines 690-730 backspace one letter if ASC(X$) = 8 and decrease the location variable (P) and the counter (A) to match the backspace.

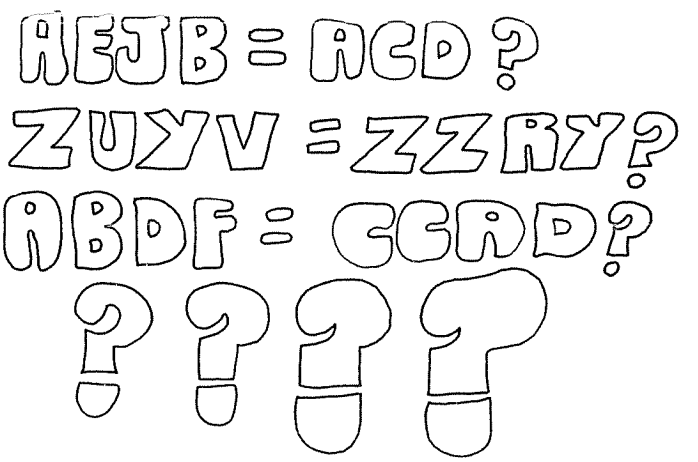
Lines 750-770 accumulate the total message entered by user.

Lines 900-1130 test for correct message entry. If the entry is correct the counter, CR, is advanced by one and another message will be selected (if any remain). If it is incorrect, the user will have the opportunity to try the same message or skip it and go to another one.

Lines 1200-1300 print the final messages and terminates the program.

Lines 4000-4100 were explained under line 230 above.

Lines 5000-5190 are the correct messages.

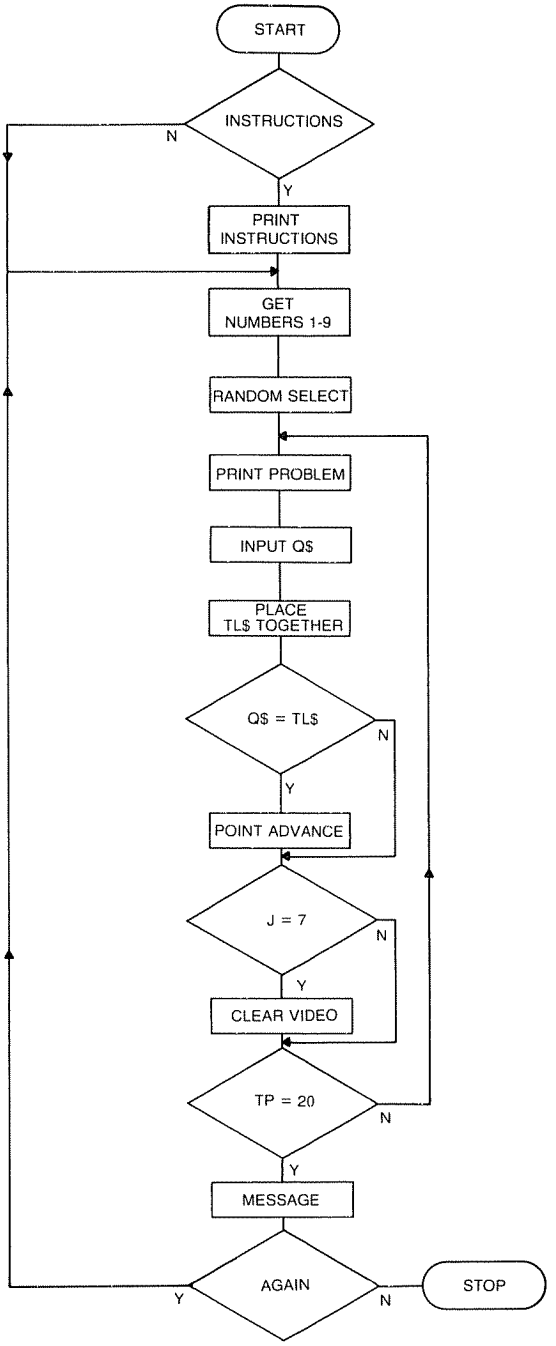


MATH-A-CODE

This program is similar to Words-Zit. Each problem you receive will have the numerals encoded as letters of the alphabet. Read all the instructions carefully before playing, noting exactly what letters represent the 0s (zeros) in any answer. It will mean the difference between winning and losing!

Sample Run

```
MATH-A-CODE

THE INSTRUCTIONS? YES

IN THESE MATH PROBLEMS YOU WILL MULTIPLY TWO
NUMBERS, THEN ADD ANOTHER TO THE ANSWER—
SIMPLE, BUT HAVE YOU TRIED IT LATELY USING LETTERS
FROM THE ALPHABET?
EACH LETTER FROM THE ALPHABET WILL REPRESENT A
NUMBER FROM 1-9 AS YOU WILL SEE IN A SAMPLE. EACH
NUMBER WILL ONLY BE USED ONCE IN EACH PROBLEM.

PRESS ENTER?

STUDY THIS SAMPLE:
HDJ * FGC + BIE = EBFFAC.
THE ANSWER IN THE ABOVE SAMPLE CONTAINS A 0
```



Flowchart for *Math a Code*.

(ZERO). THIS IS REPRESENTED BY THE LETTER A ANY-TIME YOUR ANSWER CONTAINS A 0 (ZERO), IT WILL BE THE LETTER BEFORE THE FIRST LETTER IN THE SET; I.E., IF THE FIRST LETTER IN A SET IS R, THE ZERO, IF NEEDED, WOULD BE THE LETTER Q.

PRESS ENTER?

REMEMBER, ALL LETTERS WILL REPRESENT THE NUMERALS 1-9 ONLY ONCE IN EACH PROBLEM. YOU WILL HAVE A SET OF 20 PROBLEMS TO COMPLETE ONCE YOU BREAK THE CODE, THINGS WILL GO QUITE SMOOTHLY

PRESS ENTER TO BEGIN?

HED*GCJ + BFI = ?AEFGRT

XSW*QUY + RVT = ?RFTAEE

SRL*PTN + MOQ = ?UJYTT

MLK*JGI + NHF = ?M

QTR*PSW + UOV = ?RFTTCV

VQU*ORP + TNS = ?DEAABC

FGH*LIJ + KDE = ?CXXVFG

GHF*ECJ + DBI = ?GTTFD

MJN*ILG + KHF = ?HAACD

FIL*JEH + DKG = ?RRTTG

TON*PRS + QML = ?HAAES

FEJ*GIB + CHD = ?XXSZD

NLK*MFG + IHJ = ?HTFFD

XUR*SWT + YVQ = ?JHHASD

HJE*FCB + DGI = ?VFRRA

FDI*EHJ + GLK = ?ACDDB

STR*POV + UQW = ?WAZXD

XSR*TYV + UWQ = ?BGTTD

PSN*RMK + LOQ = ?EFFAC

CGH*IKE + JFD = ?ERRGH

TOTAL POINTS FOR CORRECT
ENTRIES ARE: 0

YOU ARE READY TO TRY

20 MORE PROBLEMS, RIGHT? NO

NATURAL BORN QUITTER!!!

All answers were missed on purpose so as to not reveal the code.

Program Listing

```
10 REM PROGRAM TITLE: MATH-A-CODE
20 CLS:TP=0
30 PRINT"MATH-A-CODE"
40 INPUT"THE INSTRUCTIONS";X$
50 IF X$="N" OR X$="NO" THEN 300
60 PRINT"IN THESE MATH PROBLEMS"
65 PRINT"YOU WILL MULTIPLY"
70 PRINT"TWO NUMBERS, THEN ADD ANOTHER"
75 PRINT"TO THE ANSWER--SIMPLE. BUT, HAVE"
80 PRINT"YOU TRIED IT LATELY USING LETTERS"
85 PRINT"FROM THE ALPHABET?"
90 PRINT"EACH LETTER  FROM THE ALPHABET"
95 PRINT"WILL REPRESENT A NUMBER FROM"
100 PRINT"1-9, AS YOU WILL SEE IN A SAMPLE."
105 PRINT"EACH NUMBER WILL ONLY BE USED"
110 PRINT"ONCE IN EACH PROBLEM."
115 PRINT
120 INPUT"ENTER";X$:CLS
```

```
125 PRINT"STUDY THIS SAMPLE:"
130 PRINT"HDJ * FGC + BIE = EBFFAC"
135 PRINT"THE ANSWER IN THE ABOVE SAMPLE"
140 PRINT"CONTAINS A 0 (ZERO). THIS IS"
145 PRINT"REPRESENTED BY THE LETTER 'A'."
150 PRINT"ANYTIME YOUR ANSWER CONTAINS"
155 PRINT"A 0 (ZERO), IT WILL BE THE"
160 PRINT"LETTER BEFORE THE FIRST LETTER"
165 PRINT"IN THE SET; I.E., IF THE"
170 PRINT"FIRST LETTER IN A SET IS 'R'"
175 PRINT"THE ZERO, IF NEEDED, WOULD"
180 PRINT"BE THE LETTER 'Q'."
185 PRINT
190 INPUT"PRESS ENTER";X$:CLS
195 PRINT"REMEMBER, ALL LETTERS WILL "
200 REPRESENT THE NUMERALS 1-9"
205 PRINT"ONLY ONCE (IN EACH PROBLEM)."
210 PRINT"YOU WILL HAVE A SET OF 20"
215 PRINT"PROBLEMS TO COMPLETE IN"
220 PRINT"CODED FORM. ONCE YOU BREAK"
225 PRINT"THE CODE, THINGS WILL GO"
230 PRINT"QUITE SMOOTHLY..."
235 PRINT
240 INPUT"PRESS ENTER TO BEGIN";X$
300 CLS:Y=65:J=0:' GET NUMBERS 1-9
310 FOR I=1 TO 9:N(I)=I:NEXT
320 I=1
330 N=RND(9)
340 IF N(N)=0 THEN 330
350 A(I)=N:N(N)=0:L(I)=I
360 I=I+1
370 IF I=10 THEN 390
380 GOTO 330
390 ' PROBLEM (LETTER FORM)
400 FOR I=1 TO 3:A$=A$+STR$(A(I))
410 GOSUB 470:NEXT:PRINT" * ";
420 FOR I=4 TO 6:A1$=A1$+STR$(A(I))
430 GOSUB 470:NEXT:PRINT" + ";
440 FOR I=7 TO 9:A2$=A2$+STR$(A(I))
450 GOSUB 470:NEXT:PRINT" = ";
460 GOTO 500
470 PRINT CHR$(A(I)+Y);
480 RETURN
500 ' USER ENTRY
510 INPUT Q$
520 ' TOTAL
530 T=VAL(A$)*VAL(A1$)+VAL(A2$)
540 T$=STR$(T)
```

```
550 ' TOTAL TO STRING
560 FOR I=2 TO LEN(T$)
570 TL$=TL$+CHR$(VAL(MID$(T$,I,1))+Y)
580 NEXT
590 ' COMPARE
600 IF Q$<>TL$ THEN 620
610 PI=PI+INT(T/100):' POINTS
620 TP=TP+1:' TOTAL PLAYS
630 J=J+1:IF TP=20 THEN 800
640 A$="":A1$="":A2$=""
650 T$="":TL$=""
660 Y=RND(80)
670 IF Y<=64 THEN 660
680 PRINT
690 IF J=7 THEN 300
700 GOTO 310
800 ' MESSAGE / FINAL
810 CLS
820 PRINT"TOTAL POINTS FOR CORRECT"
830 PRINT"ENTRIES ARE: ";PI
840 PRINT
850 PRINT"YOU ARE READY TO TRY"
860 PRINT"20 MORE PROBLEMS, RIGHT";
870 INPUT X$
880 IF X$="N" OR X$="NO" THEN 900
890 TP=0:J=7:GOTO 640
900 PRINT
910 PRINT"NATURAL BORN QUITTER !!!"
920 END
```

Variable and Strings

TP - General counter, total problems
X$ - General input string
Y - ASCII contents, for coding
J - Simple counter for clearing of video, when the number of problems reaches 7
N(I) - Numerals 1-9 for coding
A(I) - Random value as above for coding
A$,A1$,A2$ - STR$ contents of code for totals
Q$ - User input of answer in letter form
T - The total value of A1, A1$, and A2$
T$ - STR$ of T (above)
TL$ - For total in letter (code) form
PI - Point total

Explanation of the Program Lines

Line 20 initializes the variable TP for the count of the total problems.

Lines 30-240 display the instructions for play.

Line 300 initializes the variables Y and J.

Lines 310-380 set the subscripted variable N(I) to the numerals 1-9; then selects the numerals 1-9 in a random fashion and places them in subscripted variable A(I), and finally deletes the number selected.

Lines 390-480 print the problem in its coded form.

Lines 500-510 allow the user to input the answer in letter form.

Lines 520-580 put together the answer in coded letter form. This is done after the user has entered his or her answer so no cheating can take place.

Lines 590-600 compare answers.

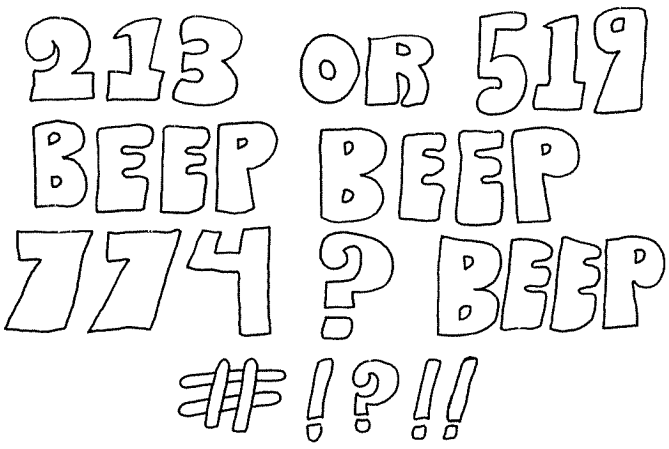
Line 610 increases the score if the answer is correct.

Line 620 advances problem counter by one.

Line 630 increases the problem count by one. If J is equal to 7, the video is cleared (line 690). If TP is equal to 20 the program branches to final message.

Lines 640-700 sets the strings to null, and get another amount for Y, and then recycle back for another problem.

Lines 800-920 print the total points accumulated and ask if the user wants another set. If he does not, the program is terminated. If he does, variables J and TP are initialized and the program recycles to the beginning.

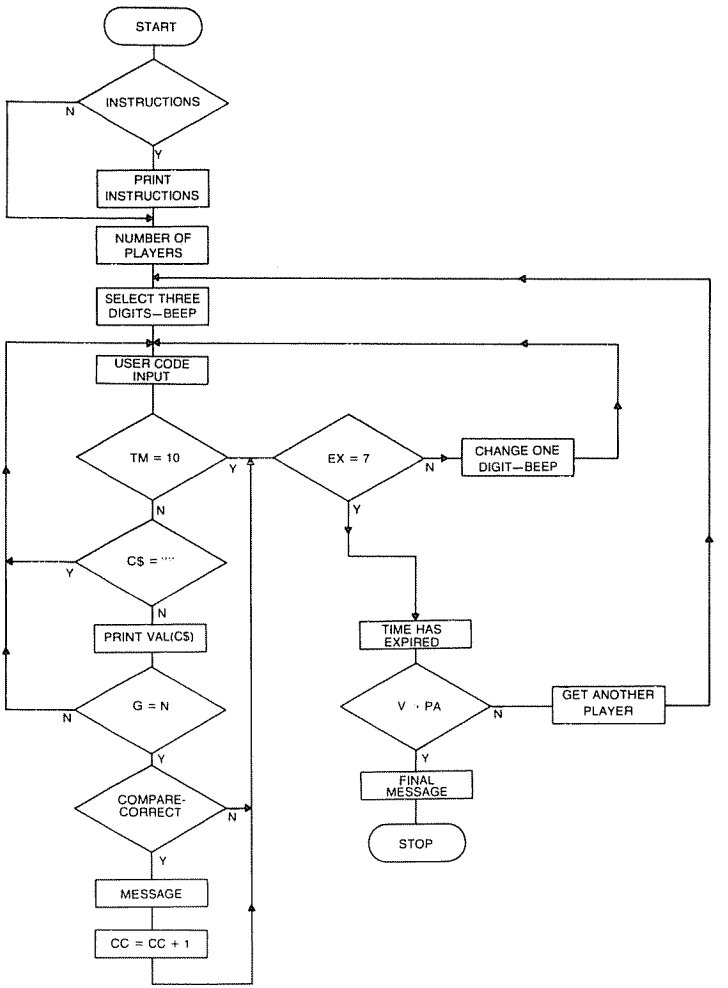


MASTER CODE

Have you ever seen so many code programs? Using the sound routine, the computer will select three digits from the numerals 1-9. The computer will then beep each digit. Each tone will represent a specified digit. You will enter your guess of the 3 digits using the INKEY$ function. The computer will beep each of your guesses as you press the numeral key. Match the tones and you've got the code, but watch out, you'll only have a certain amount of time before computer changes one of the digits! Remember to use the proper value for MC in line 370. Refer to page 3.

Sample Run

```
:: MASTER CODE ::

INSTRUCTIONS? YES

READ THE INTRODUCTORY PARAGRAPH
IN THE BOOK. REMEMBER THAT THE
COMPUTER WILL HOLD A MASTER
CODE FOR ONLY SHORT PERIODS OF
TIME AND THEN CHANGE ONE DIGIT
OF IT. AFTER A LENGTH OF TIME, IF
YOU DO NOT BREAK THE CODE,
YOU'LL BE OUT OF THE GAME.
LISTEN FOR THE THREE BEEPS:
THESE WILL BE YOUR CLUE TO
```



Flowchart for *Master Code*.

THE MASTER CODE. NUMBERS IN
THE CODE ARE 1-9 (3 DIGITS).

NUMBER TO PLAY? 2

FIRST NAMES OR INITIALS
OF THESE 2 PLAYERS? E.F.
G.T.

E.F. WILL PLAY NOW, LISTEN
TO THE THREE TONES.

Program Listing

```
10 REM PROGRAM TITLE: MASTER CODE
20 CLS
30 PRINT":: MASTER CODE ::"
40 RANDOM
50 INPUT"INSTRUCTIONS";X$
60 IF X$<>"YES" THEN 250
70 PRINT
80 PRINT"READ THE INTRODUCTORY PARAGRAPH"
90 PRINT"IN THE BOOK. REMEMBER THAT THE"
100 PRINT"COMPUTER WILL HOLD A MASTER"
110 PRINT"CODE FOR ONLY SHORT PERIODS OF"
120 PRINT"TIME AND THEN CHANGE ONE DIGIT"
130 PRINT"OF IT. AFTER A LENGTH OF TIME,"
140 PRINT"IF YOU DO NOT BREAK IT, YOU'LL"
150 PRINT"BE OUT OF THE GAME."
160 PRINT"LISTEN FOR THE THREE BEEPS,"
170 PRINT"THESE WILL BE YOUR CLUE TO"
180 PRINT"THE MASTER CODE. NUMBERS IN"
190 PRINT"THE CODE ARE 1-9 (3 DIGITS)."
250 INPUT"NUMBER TO PLAY (1-4)";PA
260 IF PA<1 OR PA>4 THEN 250
270 PRINT"FIRST NAMES OR INITIALS"
280 PRINT"OF THESE";PA;"PLAYERS";
290 FOR I=1 TO PA:INPUT N$(I)
300 NEXT
310 V=1
320 N$=N$(V):PRINT
330 PRINT N$;" WILL PLAY NOW. LISTEN"
340 PRINT"TO THE THREE TONES."
350 REM NUMBERS IN CODE
360 REM LOCATION FOR BEEPS
```

```
370 N=3:MC=31266
380 REM TOTAL NUMBERS IN SET
390 FOR I=1 TO 9:N(I)=I:NEXT
400 FOR X=1 TO 1500:NEXT:REM GET CODE
410 X=1:P=95:PL=102:D=0
420 A=INT(RND(0)*9)+1
430 IF N(A)=0 THEN 420
440 A(X)=A:N(A)=0:POKE MC,A*A+140
450 U=USR(0):X=X+1
460 IF X>N THEN CLS:GOTO 480
470 GOTO 420
480 M=0:T=0:REM BEGIN TIME
490 L=64:G=0:IF M=1 THEN 510
500 PRINT@0,"THE CODE ?";
510 C$=INKEY$
520 T=T+1:TM=INT(T/20)
530 PRINT@15,"TIME: ";TM;
540 IF C$="" THEN 600
550 H=VAL(C$):POKE MC,H*H+140
560 U=USR(0):PRINT@L,H;:L=L+2
570 G=G+1:H(G)=H:IF G=N THEN 620
580 REM TOTAL TIME
600 IF TM<>10 THEN 510
610 GOTO 720
620 REM COMPARE
630 FOR W=1 TO N
640 IF H(W)<>A(W) THEN 670
650 NEXT
660 GOTO 1000
670 PRINT@L+64-N*2,"INCORRECT";
680 FOR X=1 TO 500:NEXT:T=T+6
690 PRINT@L+64-N*2,STRING$(9,32);
700 PRINT@L-N*2,STRING$(N*2,32);
710 GOSUB 820:M=1:GOTO 490
720 PRINT@21,"* EXPIRED *";
730 REM CHANGE RANDOM NUMBERS
740 Y=RND(N)
750 EX=EX+1:IF EX=7 THEN 950
760 A=INT(RND(0)*9)+1
770 IF N(A)=0 THEN 760
780 A(Y)=A:N(A)=0:POKE MC,A*A+140
790 U=USR(0):GOSUB 1160
800 PRINT@21,STRING$(11,32);
810 PRINT@L-N*2,STRING$(N*2,32);:GOTO 480
820 REM ATTEMPTED SEQUENCES
830 IF M=0 THEN 850
840 GOTO 860
850 PRINT@35,"ATTEMPTED SEQUENCES:";
860 FOR W=1 TO N:PRINT@P,H(W);
```

```
870 P=P+2:NEXT
880 P=P-N*2+64
890 IF P>=999 THEN 910
900 RETURN
910 P=PL
920 PL=PL+N*2+1
930 D=D+1:IF D=4 THEN 950
940 GOTO 900
950 REM TIME EXPIRED (TOTAL)
960 PRINT:IF D=4 GOSUB 1100
970 PRINT"YOUR TOTAL TIME HAS EXPIRED, ";N$
980 PRINT"TOTAL CODES BROKEN =";CC
990 GOTO 1200
1000 REM CODE BROKEN
1010 PRINT
1020 PRINT"YOU'VE BROKEN THE MASTER CODE !!"
1030 PRINT"FANTASTIC, ";N$;"!! NOW TRY"
1040 PRINT"A NEW CODE..."
1050 CC=CC+1
1060 FOR X=1 TO 1000:NEXT
1070 GOTO 350
1100 REM PRINT
1110 FOR X=1 TO 15
1120 PRINT
1130 NEXT
1140 PRINT@0,;
1150 RETURN
1160 PRINT@L+135-N*2,"1 DIGIT CHANGED";
1170 FOR X=1 TO 900:NEXT
1180 PRINT@L+135-N*2,STRING$(16,32);
1190 RETURN
1200 REM ANOTHER
1210 Q(V)=CC:CC=0:GOSUB 1300
1220 V=V+1:EX=0
1230 IF V>PA THEN 1400
1240 FOR X=1 TO 1200:NEXT
1250 GOTO 320
1300 REM BEEPS (FINAL)
1310 FOR X=1 TO 20
1320 POKE MC,A(1)*A(1)+RND(200)
1330 U=USR(0):NEXT
1340 RETURN
1400 PRINT
1410 PRINT"ALL HAVE PLAYED; SCORES"
1420 PRINT"ARE AS FOLLOWS:"
1430 FOR I=1 TO PA
1440 PRINT N$(I);"'S SCORE";Q(I);
1450 PRINT"CODES BROKEN..."
```

```
1460 NEXT
1470 PRINT
1480 PRINT"END OF MASTER CODE"
1490 END
```

Variables and Strings

PA - Number of players (1-4)
N$(I) - Names of players
V - Counter for (total) number of players, PA
N$ - (Name of player) contents of N$(V)
N - Total numbers in code
MC - Location of machine language program (sound)
N(I) - Numerals 1-9 for code
X -Counter, general purpose time loop
P & PL - Print @ locations for user's code entry
D - Counter, general purpose
A - Random number select
A(X) - Numbers in code (1-3)
U - USR call for machine language routine
M - General purpose 0/1 variable for print statements
T - Time counter
TM - Total time: INT of T divided by 20
L - Print @ location for messages
G - user entry, counter of digits entered
H - Value of C$ (user entry)
H(G) - Digits selected (user entry)
W - for the comparison of digits
Y - selection (random) of a digit to be replaced
EX - for termination of program, all digits used
CC - Correct entered codes
Q(V) - Total correct codes, current user (player)

Explanation of the Program Lines

Lines 50-190 display the general instructions.

Lines 250-260 allow the user to input the number of players.

Lines 270-300 allow the input of the first name or initials of each player.

Lines 310-340 display a message that indicates who is playing.

Line 370 sets the variable that indicates the total number of digits to three, and set MC to the location that will be poked for sound.

Line 390 assigns the numerals 1 through 9 to the N(I) array.

Lines 410-470 get the three digits, and then sound a different tone for each.

Lines 480-490 initialize more variables.

Lines 500-600 let the user enter his or her selection of digits, output a tone for each selection, and test the amount of time remaining.

Line 610 branches to line 720, if TM is equal to 10.

Lines 620-650 compare each digit entered against digits selected in memory. Line 660 branches to 1000 if all are correct.

Lines 670-710 present the message if the response was incorrect.

Lines 720-810 delete one digit and select another, beep its code tone, and then it returns to line 480, if EX is not equal to 7 (which means all digits have been used).

Lines 820-940 print all three digits the user has entered that did not work. They also advance or reset the PRINT @ areas for these codes.

Lines 950-990 display the message indicating that time has expired.

Lines 1000-1070 display the message indicating that the code has been broken, advance the counter CC (codes broken) by 1, and then return to get another code if variable EX is not yet equal to 7.

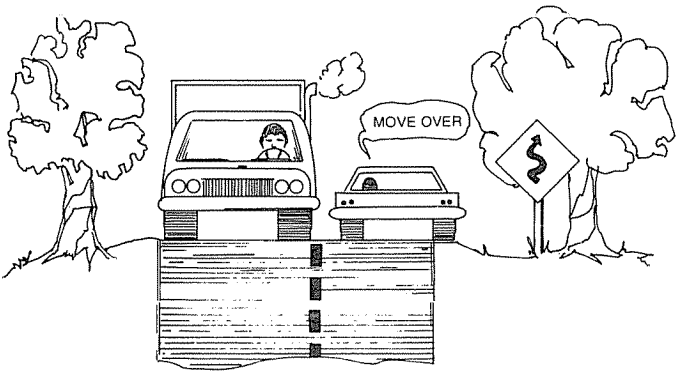
Line 1100-1150 clear the video with print statements (blanks) in preparation for the termination of program.

Line 1160-1190 display the message indicating that one of the three digits changed.

Lines 1200-1250 advance the next player, if all have not yet played (V<PA)

Lines 1300-1340 sound random beep tones that indicate the end of the program.

Lines 1400-1490 display the total scores for all players.

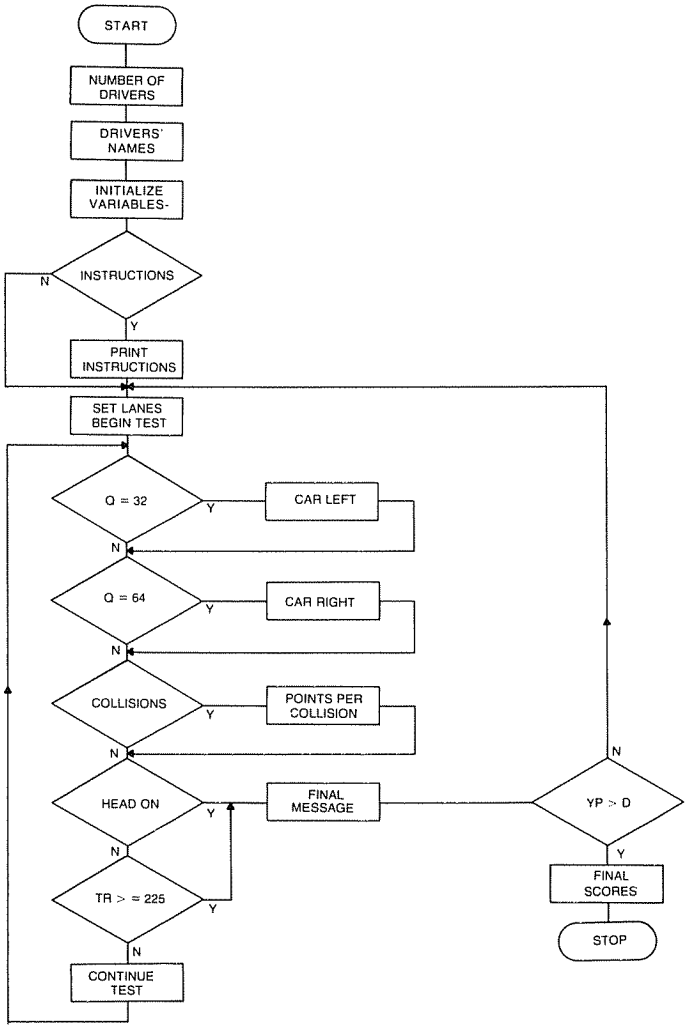


DRIVER'S TEST

Are you a good driver? Can you stay in your own lane or have enough sense to pass safely? Up to 10 people can try their ability at the wheel without leaving the house. Of course, the steering wheel will consist of the left and right arrow keys. Watch out for head-on collisions, which occur rapidly, almost without warning. The test will conclude after a determined amount of time .....or before.

Sample Run

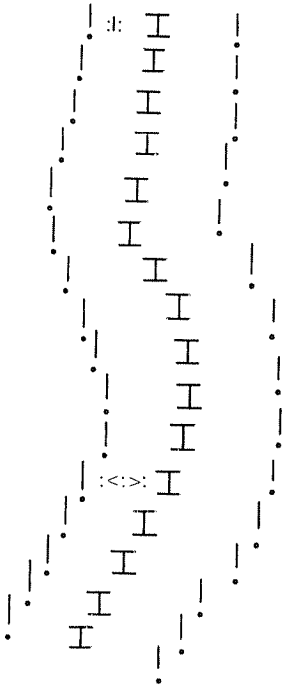
```
HOW MANY DRIVERS? 2
FIRST NAMES
? ARNOLD
?WALT

THANK YOU...
DO ANY OF YOU NEED THE
INSTRUCTIONS FOR DRIVERS TEST? YES

YOU WILL TAKE THE DRIVERS SEAT
IN AN ATTEMPT TO COME OUT ALIVE.
YOU WILL BE TRAVELLING DOWN A
ROAD, THAT WILL BE
STRAIGHT OR WINDING, YOU WILL
BE TESTED FOR HOW WELL YOU CAN
STAY IN YOUR OWN LANE: MEANING,
DON'T HIT THE SHOULDER OR THE
CENTER MEDIAN (UNLESS PASSING).
```



Flowchart for *Driver's Test*.

PRESS ENTER?

WHEN A CAR COMES UP IN FRONT OF YOU, YOU MUST PASS, TO DO THIS PRESS AND HOLD THE RIGHT ARROW KEY. AFTER PASSING QUICKLY MOVE BACK TO YOUR LANE BY PRESSING AND HOLDING THE LEFT ARROW KEY. THESE KEYS ARE ALSO USED TO KEEP YOUR CAR STEADY IN YOUR LANE. IF YOU ARE PASSING AND ANOTHER CAR IS ALREADY IN THE PASSING LANE...TAKE TO THE DITCH!!!
A PERFECT SCORE IS 0 (ZERO) MISTAKES OR ERRORS.
ARNOLD WILL NOW TAKE THE WHEEL
PRESS ENTER WHEN READY



Program Listing

```
10 REM PROGRAM TITLE: DRIVERS TEST
20 CLS
30 RANDOM
40 INPUT"HOW MANY DRIVERS";D
50 IF D<1 OR D>10 THEN 40
60 PRINT"FIRST NAMES"
70 FOR I=1 TO D:INPUT Q$(I)
80 NEXT:PRINT:YP=1
90 PRINT"THANK YOU..."
100 PRINT:M=0:WL=0:PA=0:TR=0:CA=0
105 IF YP>=2 THEN 430
110 PRINT"DO ANY OF YOU NEED THE"
120 PRINT"INSTRUCTIONS FOR DRIVERS TEST";
130 INPUT I$
140 IF I$="N" OR I$="NO" OR I$="" THEN 420
150 CLS
160 PRINT"YOU WILL TAKE THE DRIVERS SEAT"
170 PRINT"IN AN ATTEMPT TO COME OUT ALIVE."
180 PRINT"YOU WILL BE TRAVELLING DOWN A"
190 PRINT"ROAD THAT WILL BE"
200 PRINT"STRAIGHT OR WINDING. YOU WILL"
210 PRINT"BE TESTED FOR HOW WELL YOU CAN"
220 PRINT"STAY IN YOUR OWN LANE; MEANING,"
230 PRINT"DON'T HIT THE SHOULDER OR THE"
240 PRINT"CENTER MEDIAN (UNLESS PASSING)."
250 PRINT
260 INPUT"KEY ENTER";X
270 CLS
280 PRINT"WHEN A CAR COMES UP IN FRONT OF"
290 PRINT"YOU, YOU MUST PASS. TO DO THIS"
300 PRINT"PRESS AND HOLD THE <RIGHT ARROW>"
310 PRINT"KEY. AFTER PASSING QUICKLY MOVE"
320 PRINT"BACK TO YOUR LANE BY PRESSING"
330 PRINT"AND HOLDING THE <LEFT ARROW>"
340 PRINT"KEY. THESE KEYS ARE ALSO USED TO"
350 PRINT"KEEP YOUR CAR STEADY, IN YOUR"
360 PRINT"LANE. IF YOU ARE PASSING AND"
370 PRINT"ANOTHER CAR IS ALREADY IN THE"
380 PRINT"PASSING LANE...TAKE TO THE"
390 PRINT"DITCH !!!"
400 PRINT"A PERFECT SCORE IS 0 (ZERO)"
410 PRINT"MISTAKES OR ERRORS."
420 PRINT Q$(1);" WILL NOW TAKE THE WHEEL"
430 PRINT"PRESS ENTER WHEN READY";
440 INPUT X
```

```
450 CLS
500 REM TWO LANES / MEDIAN
510 R1=30:R2=44:C5=R1+3:M1=((R2-R1)/2)+R1
520 REM STRAIGHT OR CURVE / CAR
530 M3=M1:GOTO 700
540 C=RND(2):C1=RND(2):M=((R2-R1)/2)+R1
550 PRINT@C5,":I:";:IF ABS(C-C1)=0 THEN 580
560 REM STRAIGHT
570 TR=TR+1:GOTO 700
580 REM LEFT OR RIGHT CURVES
590 C2=RND(2)
600 IF ABS(C1-C2)=1 THEN 650
610 REM LEFT
620 IF ABS(R1-M1))=19 THEN 670
630 R1=R1-1:R2=R2-1:M=M-1
640 GOTO 700
650 REM RIGHT
660 IF ABS(R2-M1))=19 THEN 630
670 R1=R1+1:R2=R2+1:M=M+1
680 REM LANE AND A CAR
700 REM USER CAR MOVEMENT
710 Q=PEEK(15340)
720 IF R1)=960 OR R2)=960 THEN 740
730 R1=R1+64:R2=R2+64:M1=M1+64
740 PRINT@R1,"!";:PRINT@C5,":I:";
750 PRINT@M,"I";:PRINT@R2,"!";
755 IF PA=3 THEN CA=CA-64
756 IF CL=1 THEN CC=CC-64
757 IF ABS(CA-C5)(=3 THEN 1160
760 IF M=0 PRINT@M,"  ";
770 IF Q=32 THEN 810
780 IF Q=64 THEN 830
790 PRINT:IF TR)=225 THEN 970
792 IF ABS(CC+64-C5)(=3 THEN 1220
795 IF PA(>3 AND C5>M3 AND Q(>32 THEN 870
800 GOTO 900
810 REM CAR LEFT
812 IF C5(=0 THEN ZT=1:GOTO 1220
815 IF PA=3 THEN C5=C5-2:GOTO 790
820 C5=C5-1:GOTO 790
830 REM CAR RIGHT
832 IF C5)=62 THEN ZT=1:GOTO 1220
835 IF PA=3 THEN C5=C5+2:GOTO 850
840 C5=C5+1
850 IF PA(>3 AND C5>M3+2 THEN 870
860 GOTO 790
870 REM STAYING IN WRONG LANE
880 WL=WL+1:GOTO 1000
```

```
900 Q1=PEEK(C5+15360+64)
910 Q2=PEEK(C5+15360+66)
920 IF Q1<>32 AND Q<>32 THEN 940
925 IF Q2<>32 AND Q<>32 THEN 940
930 IF ABS(M3-C5))=14 THEN 960
935 GOTO 1000
940 IF PA=3 THEN 1160
950 REM WRONG LANE / SHOULDER
960 WL=WL+1:GOTO 1000
970 PRINT@0,"DRIVING TEST COMPLETED..."
980 GOTO 1460
1000 REM CAR TO PASS
1001 IF PA=3 THEN 1040
1010 PA=RND(20)
1020 IF PA<>1 THEN 1330
1030 PA=PA+2
1040 IF R1<=860 THEN PA=0:GOTO 540
1050 IF CA>=100 THEN 1070
1060 IF CP=0 THEN CA=M-6
1070 IF CP=1 THEN 1110
1080 PRINT@CA,":<I>:";
1090 CP=1
1100 GOTO 540
1110 REM CLEAR IF LESS THAN USER CAR
1120 IF CA<=0 THEN 1140
1130 GOTO 540
1140 PA=0:CP=0:CL=0
1150 GOTO 1190
1160 REM HEAD-ON ?
1170 IF ABS(CA-C5)<=4 THEN 1220
1180 IF ABS(CA+1-C5)<=3 THEN 1220
1190 IF PA+CP<>4 AND C5<R1-962 THEN 960
1200 IF PA+CP<>4 AND C5>R2-962 THEN 960
1210 GOTO 540
1220 X=1:WL=-99
1230 PRINT@C5,"*";:PRINT@C5+2,"*";
1240 PRINT@C5+64-1,"*";:PRINT@C5+64+4,"*";
1250 FOR U=1 TO 50:NEXT
1260 PRINT@C5,"      ";:PRINT@C5+64+1,"*";
1270 PRINT@C5+64-3,"         ";
1280 FOR U=1 TO 25:NEXT
1290 X=X+1
1300 IF X>=11 THEN 1400
1310 GOTO 1230
1330 REM CAR IN OPPOSITE LANE
1340 IF M<896 OR PA<>10 AND PA<>15 THEN 540
1350 IF CL=1 THEN 540
1360 CL=1:CC=M+2
1370 PRINT@M+2,":--:";
```

```
1380 GOTO 540
1400 REM FINISHED / DECEASED
1410 PRINT@0,;
1420 PRINT"YOU HAVE BECOME ANOTHER"
1430 PRINT"HIGHWAY STATISTIC."
1435 IF ZT=1 THEN ZT=0:GOTO 1460
1440 PRINT"MOVE TO THE PASSING LANE"
1450 PRINT"A LITTLE SOONER...NEXT TIME."
1460 UP(YP)=WL:YP=YP+1
1470 IF YP>D THEN 1520
1480 PRINT
1490 PRINT Q$(YP);" WILL NOW TAKE THE"
1500 PRINT"WHEEL, GOOD LUCK !!!"
1510 CP=0:CL=0:CC=0:GOTO 100
1520 FOR T=1 TO 2000:NEXT:CLS
1530 PRINT"END OF DRIVING TEST."
1540 PRINT"TOTAL NUMBERS OF ERRORS"
1550 PRINT"ARE AS FOLLOWS:"
1560 FOR I=1 TO D
1570 IF UP(I)=-99 THEN 1620
1580 PRINT Q$(I);" HAD";UP(I);
1590 PRINT"TOTAL ERRORS."
1600 NEXT
1610 GOTO 1640
1620 PRINT Q$(I);" IS DECEASED..."
1630 GOTO 1600
1640 PRINT
1650 PRINT"END OF DRIVERS TEST."
1660 END
```

Variables and Strings

D - Number of drivers
Q$(I) - First names of the D drivers
YP - Players advance (counter)
WL - Counter, staying in wrong lane
PA - For use with alternate car
CA - Alternate car
R1 - Left side of lane
R2 - Right side of lane
Q1 & Q2 - Test for collisions
C5 - User's car location
M & M1 & M3 - Center lane (median)
C & C1 - Determine if straight or curved lanes will be used
TR - Length of driving test, terminates when TR is greater than or equal to 225
C2 - Left or right curve

Q - User's key closure
CL - Car already in opposite lane
CC - Alternate car location
UP(YP) - Current player's score, UP(YP) = – 99 denotes that player is deceased (had head-on collision)

Explanation of the Program Lines

Lines 10-450 determine the number of drivers, initialize several of the variables, and display the general instructions.

Lines 500-670 set the variables for lane location and left or right curves.

Lines 689-750 test for user's car movement and print the two lanes down the video screen.

Lines 755-757 print the other car and checks for a pass or a collision.

Line 760 clears the center median.

Lines 770-780 branch to the appropriate lines for the user's left or right driving movements.

Line 790 includes a print statement that keeps the lanes jumping to simulate car movement and tests for a completed drivers test by determining whether or not TR is greater than or equal to 225.

Line 792 checks for a collision.

Line 795 tests the user's car in case he or she is staying in wrong lane.

Lines 810-820 manipulate the variables to move the user's car to the left.

Lines 830-860 manipulate the variables to move the user's car to the right.

Lines 870-960 increase the counter for the user staying in the wrong lane and also test for collisions with the other car.

Lines 970-980 display the message that the test is over.

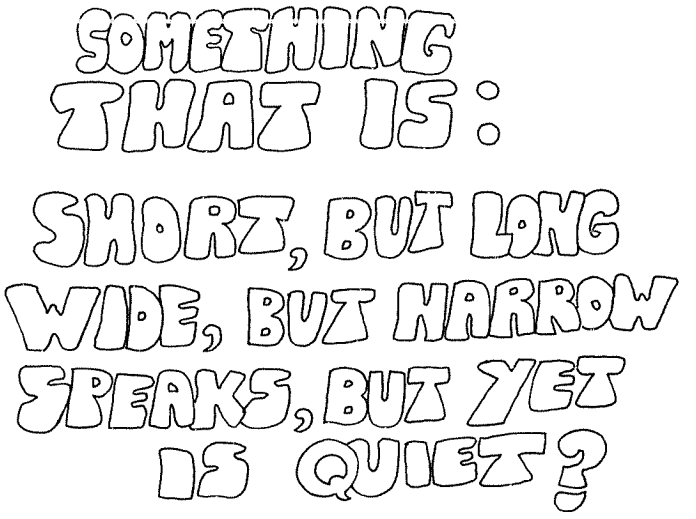
Lines 1000-1100 will print another car at bottom of video if the arguments are met, and there is not already another car at that location.

Lines 1110-1150 clear the appropriate variables of the other car if the location of that car is less than the present location of the user's car.

Lines 1160-1310 test for a head-on collision with the other car, demolish both if conditions are met, and branch to a message to go to the next driver.

Lines 1330-1380 print a car in the opposite lane (the third car).

Lines 1400-1660 advance to the next driver (player), print all scores, and terminate the program if conditions are met.



WHAT IS IT?

You just might be saying "What is it" to yourself after completing the game! The computer will give you five different clues to a selected word. Your score will be determined by how many guesses it takes you to figure out the word. Up to four people can beat their brains out with this one.

Sample Run

```
---WHAT IS IT?---
YOU WILL HAVE UP TO FIVE
CLUES TO HELP YOU FIGURE OUT WHAT WORD THE
COMPUTER HAS STORED IN ITS
MEMORY. THE PLAYERS CAN TAKE
AS MANY GUESSES AS THEY WISH,
BUT THE MORE GUESSES IT TAKES
YOU, THE LESS YOU'LL WIN.
YOUR SCORE FOR EACH WORD STARTS
AT 50; THEN 10 POINTS ARE DEDUCTED
FOR EACH NEW CLUE.
EACH ITEM CONTAINS
ONLY ONE WORD; I.E. 'A SWORD'
WOULD JUST BE INPUT AS
'SWORD'....
```

NUMBER TO PLAY? 2
FIRST NAMES OF THESE 2 PLAYERS, PLEASE
?FRED
?YIZZY

FRED WILL PLAY.....

FRED THIS IS CLUE NUMBER 1
TO DESTROY
WHAT IS THE WORD? GET RID OF

FRED THIS IS CLUE NUMBER 2
REMOVE
WHAT IS THE WORD? TAKEAWAY

FRED THIS IS CLUE NUMBER 3
WEAR AWAY
WHAT IS THE WORD? RUIN

FRED THIS IS CLUE NUMBER 4
CAUSE TO DISAPPEAR
WHAT IS THE WORD? ERASE

FRED THIS IS CLUE NUMBER 5
ERASED
WHAT IS THE WORD? DESTROY
SORRY, FRED THAT WAS 5
CLUES AND YOU DID NOT
ENTER THE RIGHT WORD.

YIZZY WILL PLAY.....

YIZZY THIS IS CLUE NUMBER 1
TO USE IS TO PAY
WHAT IS THE WORD? RESTROOM

YIZZY THIS IS CLUE NUMBER 2
GOES ALMOST EVERYWHERE
WHAT IS THE WORD? CAR

YIZZY THIS IS CLUE NUMBER 3
LOTS OF NOISE

WHAT IS THE WORD? KIDS

YIZZY THIS IS CLUE NUMBER 4
TERRIFIC SPEED
WHAT IS THE WORD? RUNNER

YIZZY THIS IS CLUE NUMBER 5
SST IS ONE
WHAT IS THE WORD? JET

SORRY, YIZZY THAT WAS 5
CLUES AND YOU DID NOT
ENTER THE RIGHT WORD.

FRED WILL PLAY.....

FRED THIS IS CLUE NUMBER 1
STRIPES OR SOLID
WHAT IS THE WORD? ARMY

FRED THIS IS CLUE NUMBER 2
LARGE IS BETTER
WHAT IS THE WORD? CAR

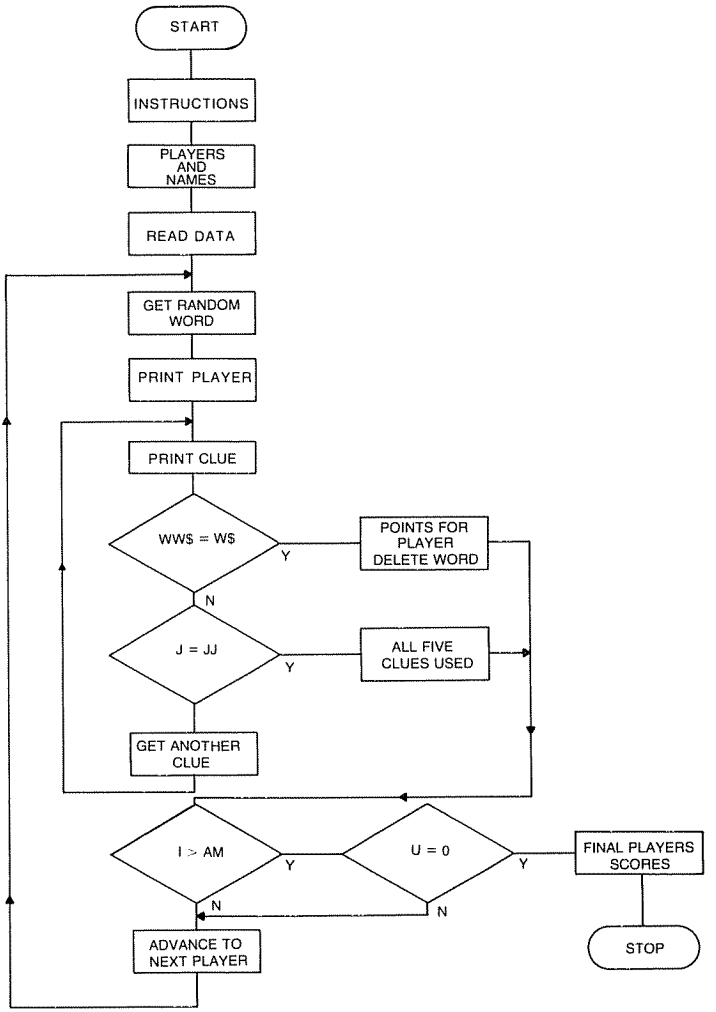
FRED THIS IS CLUE NUMBER 3
EATEN IN WARM WEATHER
WHAT IS THE WORD? FRUIT

FRED THIS IS CLUE NUMBER 4
BETTER ICE COLD
WHAT IS THE WORD? BEER

FRED THIS IS CLUE NUMBER 5
BUST IF DROPPED
WHAT IS THE WORD? BOTTLE

SORRY, FRED THAT WAS 5
CLUES AND YOU DID NOT
ENTER THE RIGHT WORD.

YIZZY WILL PLAY.....



Flowchart for *What Is It?*

YIZZY THIS IS CLUE NUMBER 1
TO DESTROY
WHAT IS THE WORD? BEAT

YIZZY THIS IS CLUE NUMBER 2
REMOVE
WHAT IS THE WORD? CARRYOUT

YIZZY THIS IS CLUE NUMBER 3
WEAR AWAY
WHAT IS THE WORD? CARRYOUT

YIZZY THIS IS CLUE NUMBER 4
CAUSE TO DISAPPEAR
WHAT IS THE WORD? FLUSH

YIZZY THIS IS CLUE NUMBER 5
ERASED
WHAT IS THE WORD? PENCIL

SORRY, YIZZY THAT WAS 5
CLUES AND YOU DID NOT
ENTER THE RIGHT WORD.

STOP

This is not a complete run.

Program Listing

```
 10 REM PROGRAM TITLE: WHAT IS IT?
 20 CLEAR 1000
 30 RANDOM:CLS
 40 DIM W$(16),A$(80),R(16)
 50 PRINT"--- WHAT IS IT? ---"
 60 FOR T=1 TO 1200:NEXT
 70 PRINT"YOU WILL HAVE UP TO FIVE"
 80 PRINT"CLUES TO HELP YOU FIGURE OUT WHAT
    WORD THE"
 90 PRINT"COMPUTER HAS STORED WITHIN IT'S"
100 PRINT"MEMORY  THE PLAYER CAN TAKE"
110 PRINT"AS MANY GUESSES AS THEY WISH,"
120 PRINT"BUT, THE MORE GUESSES IT TAKES"
130 PRINT"YOU, THE LESS YOU'LL"
140 PRINT"WIN.YOUR SCORE STARTS"
```

```
150 PRINT"AT 50.10 POINTS ARE DEDUCTED"
160 PRINT"FOR EACH NEW CLUE."
170 PRINT"EACH ITEM CONTAINS"
180 PRINT"ONLY ONE WORD, I.E. 'A SWORD'"
190 PRINT"WOULD JUST BE INPUT AS"
200 PRINT"'SWORD'...."
210 FOR T=1 TO 10000:NEXT:CLS
220 INPUT"NUMBER TO PLAY (1-4)";AM
230 IF AM<1 OR AM>4 THEN PRINT:GOTO 220
240 PRINT"FIRST NAMES OF THESE";AM;
250 PRINT"PLAYERS, PLEASE"
260 FOR I=1 TO AM
270 INPUT X$(I):NEXT
280 I=1:U=1
290 X$=X$(I):PRINT X$;" WILL PLAY....."
300 PRINT:IF U>=2 OR UU<>0 THEN 340
310 C=1:FOR A=1 TO 16:READ W$(A)
320 R(A)=A:NEXT
330 FOR A=1 TO 80:READ A$(A):NEXT
340 REM GET ONE RANDOM WORD
350 REM IF NOT DONE
360 R=RND(16)
370 IF R(R)=0 THEN 360
380 W$=W$(R):J=R*5-4:JJ=J+4:TP=50
390 PRINTX$;" THIS IS CLUE NUMBER";C
400 A$=A$(J):PRINT A$:REM A$(J) FOR CLUE
410 INPUT"WHAT IS THE WORD";WW$
420 IF WW$<>W$ THEN 440
430 R(R)=0:GOTO 560
440 IF J=JJ THEN 470
450 TP=TP-10:J=J+1:C=C+1
460 PRINT:GOTO 390
470 PRINT:TP=0
480 PRINT"SORRY, ";X$;" THAT WAS";C
490 PRINT"CLUES AND YOU DID NOT"
500 PRINT"ENTER THE RIGHT WORD."
510 IF AM=1 THEN 540
520 I=I+1
530 IF I>AM THEN I=1
540 C=1:U=U+1:IF U>=17 THEN 800
545 IF G=1 AND U>UU THEN 600
550 FOR T=1 TO 1500:NEXT:CLS:GOTO 290
560 TP(I)=TP(I)+TP:PRINT
570 PRINT"THAT IS THE WORD ";X$
580 PRINT"YOU NOW HAVE";TP(I);"POINTS."
590 GOTO 510
600 REM ALL USED
610 FOR T=1 TO 2000:NEXT
```

```
 620 CLS
 630 PRINT"ALL WORDS IN THE DATA LIST."
 640 PRINT"HAVE BEEN USED.TOTAL SCORES"
 650 PRINT"FOR THE PLAYER ARE:"
 660 FOR I=1 TO AM
 670 PRINT"POINTS FOR ";X$(I);" ARE:";
 680 PRINT TP(I)
 690 NEXT
 700 PRINT
 710 PRINT"END OF GAME RUN."
 720 END
 800 REM TEST FOR ALL USED
 810 G=1
 820 U=0:FOR T=1 TO 16
 830 IF R(T)<>0 THEN U=U+1
 840 NEXT
 850 IF U=0 THEN 600
 860 REM MORE REMAIN
 870 UU=U:U=1:GOTO 550
1000 REM WORD DATA
1010 DATA GEAR,WATERMELON,MONEY,OBLITERATE
1020 DATA YEAR,FOOD,AIRLINER,THREAD
1030 DATA TRAIN,OCEAN,PAPER,PARTY
1040 DATA BOX,CENTER,SNAKE,DOOR
1050 REM DATA CLUES
1060 DATA HAS TEETH,DIFFERENT USES
1070 DATA ROUND,MADE OF DIFFERENT MATERAL
1080 DATA IN TOYS OR MACHINES
1090 DATA STRIPES OR SOLID,LARGE IS BETTER
1100 DATA EATEN IN WARM WEATHER
1110 DATA BETTER ICE COLD,BUST IF DROPPED
1120 DATA ALL NEED IT,EXCHANGES HANDS
1130 DATA MANY USES,MADE IN ALL COUNTRIES
1140 DATA YOU CAN'T TAKE IT WITH YOU
1150 DATA TO DESTROY,REMOVE,WEAR AWAY
1160 DATA CAUSE TO DISAPPEAR,ERASED
1170 DATA CAN LEAP,CONTAINS 12 THINGS
1180 DATA HAS MORE DAYS THAN ABOVE
1190 DATA CELEBRATED AT FIRST
1200 DATA STARTS WITH MONTH ONE
1205 DATA YOU NEED IT,THOUSANDS OF VARITIES
1210 DATA TOO MUCH MAKES YOU SICK
1220 DATA MADE IN DIFFERENT WAYS
1230 DATA MOSTLY GROWN
1240 DATA TO USE IS TO PAY
1250 DATA GOES ALMOST EVERYWHERE
1260 DATA LOTS OF NOISE,TERRIFIC SPEED
1270 DATA SST IS ONE
```

```
1280 DATA LARGER THAN A HAIR,MENDS
1290 DATA HAS STRENGTH,USED TO MAKE CLOTHES
1295 DATA IN ALL COUNTRIES
1300 DATA HAS WHEELS,USES NO GAS
1310 DATA PASSENGERS,CARGO,SETS ON RAILS
1320 DATA IS NOT DRY,GOES TO MANY DEPTHS
1330 DATA CAN BE VIOLENT,CONTAINS FOOD
1340 DATA SALTY,CAN TEAR,CAN BE FOLDED
1350 DATA MADE FROM A TREE,DIFFERENT SIZES
1360 DATA LIGHT OR HEAVY
1400 DATA FRIENDS GATHER,TO CELEBRATE
1410 DATA CAKE OR DRINKS,CANDLES
1420 DATA CAN LAST ALL NIGHT,FOLDED WHEN MADE
1425 DATA LIGHTWEIGHT OR HEAVYWEIGHT
1430 DATA USED FOR SHIPPING OR STORING
1435 DATA MADE FROM TREES
1440 DATA COMPUTER CAME IN ONE
1450 DATA DIVIDES,NOT LEFT OR RIGHT
1460 DATA PERSON MOST IMPORTANT
1470 DATA OF A CIRCLE,POINT_OF_NERVES
1480 DATA HAS NO LEGS,LONG OR SHORT
1490 DATA CAN STRIKE,CAN KILL,SHEDS SKIN
1500 DATA MOST OF WOOD,HAS A LOCK
1510 DATA BLOCKS INTRUDERS,OPENS
1520 DATA AND CLOSES
```

Variables and Strings

AM - Number of players (1-4)
X$(I) - First names of players
W$(I) - Words
A$(I) - Clues
R(I) - For deleting already selected words
J,JJ - For correct clue to word selected
WW$ - User input of word
TP - Total points
C - Clue counter, up to 5
U - Counter for all words
TP(I) - Total points gained by current player

Explanation of the Program Lines

Lines 10-210 clear memory for string entries, dimensions the arrays for words and clues, and displays the general instructions.

Lines 220-270 request the number of players and their first names.

Line 280 sets the counters for the players and words.

Line 290 indicates the current player.

Line 300 branches to line 340 if the data has been read.

Lines 310-330 read all words and clues. R(A) is used for deleting already selected words.

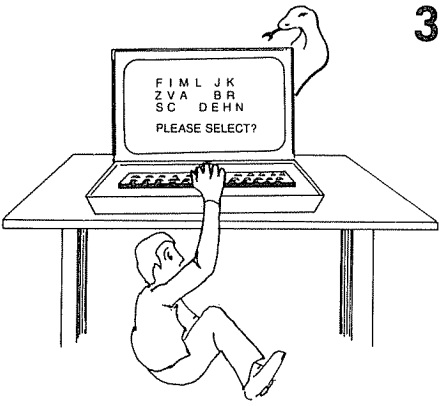
Lines 340-380 select one random word and using variables J & JJ, get its appropriate clue (line 380).

Lines 390-590 allow the current player to enter his or her word to match the clues given. If the word does not match, the total points (TP) are decreased by 10. The computer then gives another clue, until 5 clues have been presented. If the user's word does not match, he or she is given the number of points indicated by the current value of TP. That word is then deleted, and the program advanced to the next player if more than one is playing. If the current player has used all 5 clues and did not get the correct word, that word will be retained for future use, and the program again advances to the next player.

Lines 600-720 print all the player's names and their total points and then terminate the game.

Lines 800-870 test to see if all data elements (words) have been selected and deleted. If not, program flow will branch to the next player, and the computer will select one of those words. If all the words have been used, the program is terminated using lines 600-720.

Lines 1000-1520 contain all the data elements for the words and clues.



Games of General Interest

Where's the Necessary Room lets you experience drunkenness without touching a bottle. See if you can find your way through the bar with the help of an inebriated computer.

Troubles allows you to sob on the shoulders of the computer, which will offer you some sound advice.

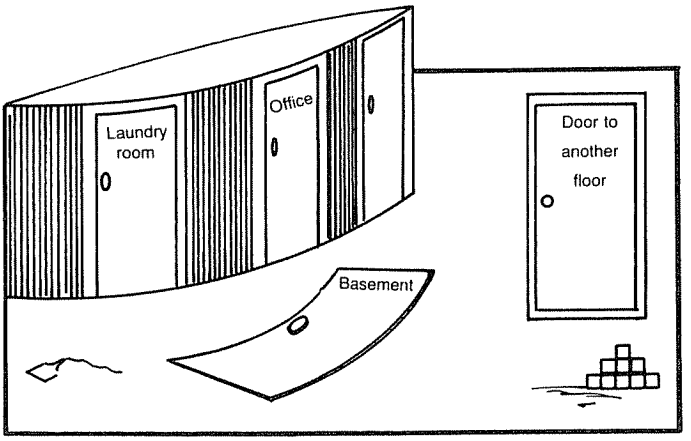
General Store pressures you to add prices together quickly. The swiftest thinker will win the game.

Respect Your Computer is a game you'll never win!

Love One Another lets you bring your problems with your spouse or friend to the computer and helps you resolve your disagreements.

Date Service helps you determine who you are most compatible with.

Key a Date allows you to keep a secret file of all those you might like to call for a date. The information is safe from prying eyes.



WHERE'S THE NECESSARY ROOM?

You're minding your own business inside this bar that is large enough to let two pro-football teams battle it out. You've imbibed quite a lot of liquid, and now you have an urge, a pain, and you must relieve it soon or "#"! So, instead of asking anyone, you and your computer will now try and locate that little room. There are plenty of meanies in this bar as well as many rooms ! Don't let the computer fool you.

Sample Run

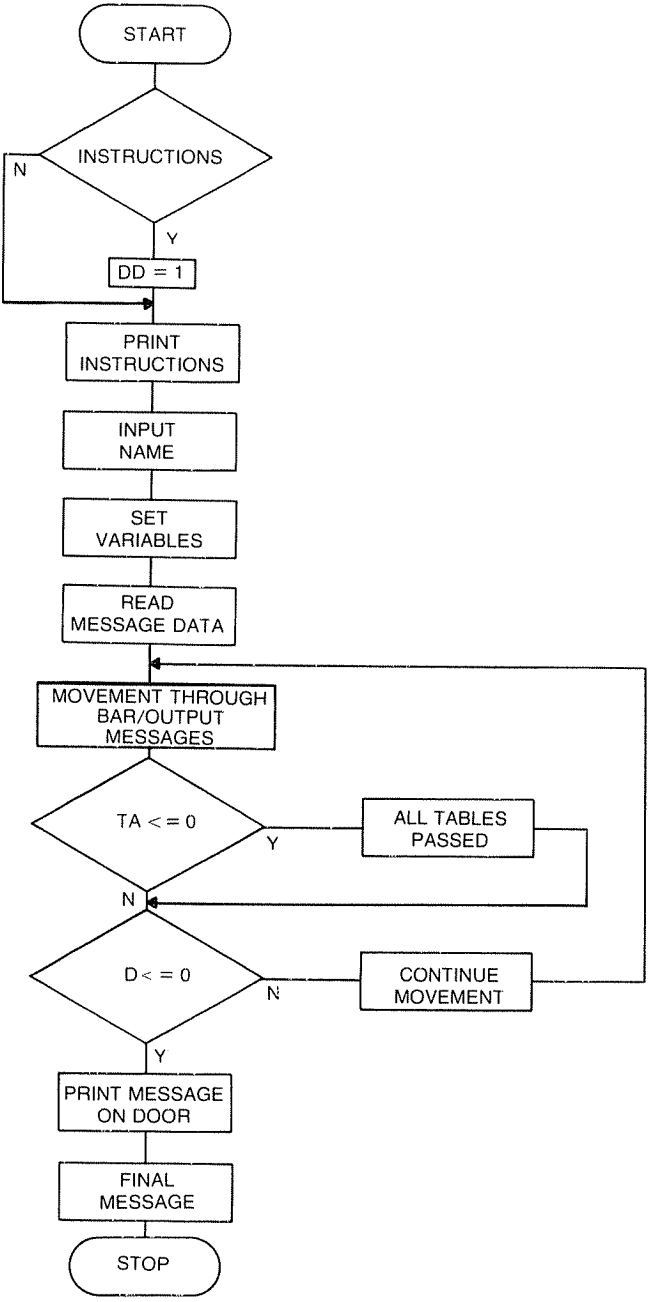
```
WHERE'S THE NECESSARY ROOM???

SEE THE INSTRUCTIONS...YOU DRUNK? YES

SO HERE YOU ARE AGAIN, YOU LUSH!!!
FILLED YOUR KIDNEYS TO THE BRINK;
YOUR HEAD IS SPINNING AND WHAT
DO YOU NEED ?? NATURALLY, THIS
BEING A COMPUTER PROGRAM, I YOUR
COMPUTER WILL TRY AND HELP YOU
LOCATE THAT LITTLE ROOM (IF THERE
IS ONE!!) STEP BY STEP.
NATURALLY THIS BEING A COMPUTER
PROGRAM (DID I SAY THAT?) I HAVE
ALSO HAD A LITTLE TOO MUCH...
```

HIT THE WHITE KEY?
YOU DRUNK!!!
AS I SAY...I AM ALSO A LITTLE
TIPSY, IF MY CIRCUITS HOLD TO...
GETHER LONG ENOUGH, WE TOGETHER
WILL FIND THAT ROOM. WHAT'S YOUR
NAME CLYDE? GRETA
YOUR KIDDING, SUCH A LONG NAME.
SO...TO FIND THAT ROOM, I WILL
INSTRUCT YOU ON THE ROUTE. YOU
IN TURN WILLL PRESS EITHER THE
LEFT ARROW KEY...OR THE RIGHT
ARROW KEY, THAT IS IF YOU
BELIEVE WHAT I TELL YOU. IF
LUSH, HIT THAT KEY AGAIN?
YOU DO NOT BELIEVE ME, PRESS
THAT LONG BAR, AT THE BOTTOM
OF THE KEYBOARD...OK GRETA?
EXCUSE ME WHILE I UNSCRAMBLE A
FEW FLIP-FLOPS.....

GRETA WE ARE NOW AT TABLE 35
MEANING, WE MUST TRAVEL 35
TABLES TO REACH THE BAR.
TOTAL DISTANCE, I WOULD ESTIMATE
TO BE AROUND 144 FEET.
GLANCING AROUND (SOMETHING YOU'RE
NOT TOO GOOD AT) I JUDGE THERE ARE ABOUT 68
CUSTOMERS WE MUST GET PAST...

OK, LUSH...I MEAN GRETA
GET UP AND MOVE RIGHT, AWAY
FROM MS. FOUL MOUTH. SHE'S A MEANIE
WE DON'T WANT TO TANGLE WITH.

PRESS THE RIGHT ARROW OR
THE LEFT ARROW OR THE LONG
BAR...YOU DRUNK, QUIT BOBBIN'
YOUR HEAD !!
OK, LUSH...I MEAN GRETA
GET UP AND MOVE LEFT, AWAY



Flowchart for *Where's the Necessary Room?*

FROM MR. SMASH. HE'S A MEANIE
WE DON'T WANT TO TANGLE WITH.

PRESS THE RIGHT ARROW OR
THE LEFT ARROW OR THE LONG
BAR...YOU DRUNK, QUIT BOBBIN'
YOUR HEAD !!
WE'RE GETTING CLOSER GRETA
HOLD THAT BLADDER !!!

OUR DISTANCE IS NOW AT 41 FEET.
WITH 17 TABLES AND
STILL 79 CUSTOMERS TO GET BY.
EXCUSE ME, THAT'S 76 CUSTOMERS.
A FEW JUST PASSED OUT...

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES
STILL 76 CUSTOMERS TO GET BY.
OK LUSH...I MEAN GRETA
ALL DIRECTIONS POINT LEFT, AWAY
FROM MR. P. WADDLE. HE'S A MEANIE,
WE DON'T WANT TO TANGLE WITH.

I SAID, GO LEFT, YOU WENT RIGHT
GRETA, YOU ARE A LITTLE MORE
DRUNK THAN I FIRST THOUGHT !!
YOU JUST STEPPED ON MR. MUZZLE TOES.
CAN'T YOU LISTEN TO DIRECTIONS ??

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 76 CUSTOMERS TO GET BY.
EXCUSE ME, THAT'S 75 CUSTOMERS.
A FEW MORE JUST PASSED OUT...

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 75 CUSTOMERS TO GET BY.
OK LUSH...I MEAN GRETA
LOOK RIGHT THEN MOVE RIGHT, AWAY FROM MS. LEAN
TOO. SHE'S A MEANIE

WE DON'T WANT TO TANGLE WITH.
WRONG DOOR !!
THIS IS THE BASEMENT.

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 75 CUSTOMERS TO GET BY.
OK, LUSH...I MEAN GRETA
LOOK LEFT THEN MOVE LEFT, AWAY
FROM MR. MUZZLE. HE'S A MEANIE
WE DON'T WANT TO TANGLE WITH.
GUESS WHAT YOU'VE JUST WALKED
INTO? YES, THE OFFICE! FEEL EMBARRASSED, YOU
DRUNK !!

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 75 CUSTOMERS TO GET BY.
OK,LUSH...I MEAN GRETA

PRESS THE RIGHT ARROW OR
THE LEFT ARROW OR THE LONG
BAR...YOU DRUNK, QUIT BOBBIN'
YOUR HEAD !!
I'M SORRY, MY MISTAKE, I TOLD
YOU THE WRONG DIRECTION GRETA.
DINGY !! YOU SHOULDN'T HAVE
LISTENED THIS TIME !!!

HOW DID THAT BOTTLE THAT MS. FOUL MOUTH
JUST BROKE OVER YOUR HEAD FEEL ??

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 75 CUSTOMERS TO GET BY.
EXCUSE ME, THAT'S 70 CUSTOMERS.
A FEW MORE JUST PASSED OUT...

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 70 CUSTOMERS TO GET BY.
OK, LUSH...I MEAN GRETA

WE WILL NOW MOVE RIGHT

I'M SORRY, MY MISTAKE, I TOLD
YOU THE WRONG DIRECTION GRETA.
YOU SHOULDN'T HAVE
LISTENED THIS TIME !!!

OF ALL THE MEANIES TO RUN IN TO !!!
NEXT TIME MS. FOUL MOUTH WILL PROBABLY
FINISHING BREAKING YOUR FACE...

OUR DISTANCE STANDS AT 41 FEET.
WITH 17 TABLES AND
STILL 70 CUSTOMERS TO GET BY.

STOP

This is not a complete run.

Program Listing

```
 10 REM PROGRAM TITLE: WHERE'S THE
 20 REM NECESSARY ROOM ???
 30 CLS
 40 PRINT"WHERE'S THE NECESSARY ROOM ???"
 50 PRINT
 60 PRINT"SEE THE INSTRUCTIONS...";
 65 FOR I=1TO500:NEXT:PRINT"YOU DRUNK";
 70 INPUT X$
 80 IF X$<>"Y" AND X$<>"YES" THEN DD=1:
    GOTO 275
 90 PRINT"SO HERE YOU ARE AGAIN, YOU
    LUSH !!!"
100 PRINT"FILLED YOUR KIDNEYS TO THE BRINK;"
110 PRINT"YOUR HEAD IS SPINNING AND WHAT"
120 PRINT"DO YOU NEED ?? NATURALLY, THIS"
130 PRINT"BEING A COMPUTER PROGRAM, I YOUR"
140 PRINT"COMPUTER WILL TRY AND HELP YOU"
150 PRINT"LOCATE THAT LITTLE ROOM (IF THERE"
160 PRINT"IS ONE!!) STEP BY STEP."
170 PRINT"NATURALLY THIS BEING A COMPUTER"
180 PRINT"PROGRAM (DID I SAY THAT?) I HAVE"
190 PRINT"ALSO HAD A LITTLE TOO MUCH..."
200 PRINT
```

```
210 INPUT"HIT THE WHITE KEY";X$
220 CLS
230 PRINT"YOU DRUNK!!!"
240 PRINT"AS I SAY...I AM ALSO A LITTLE"
250 PRINT"TIPSY, IF MY CIRCUITS HOLD TO..."
260 PRINT"GETHER LONG ENOUGH, WE TOGETHER"
270 PRINT"WILL FIND THAT ROOM. WHAT'S YOUR"
275 IF DD=1 THEN PRINT"THE ";
280 INPUT"NAME CLYDE";N$
290 IF LEN(N$)>=3 THEN 310
300 GOTO 320
310 PRINT"YOUR KIDDING, SUCH A LONG NAME."
320 PRINT:IF DD=1 THEN 460
330 PRINT"SO...TO FIND THAT ROOM, I WILL"
340 PRINT"INSTRUCT YOU ON THE ROUTE. YOU"
350 PRINT"IN TURN WILL PRESS EITHER THE"
360 PRINT"LEFT ARROW KEY...OR THE RIGHT"
370 PRINT"ARROW KEY, THAT IS IF YOU"
380 PRINT"BELIEVE WHAT I TELL YOU. IF"
390 PRINT
400 INPUT"LUSH, HIT THAT KEY AGAIN";X$
410 CLS
420 PRINT"YOU DO NOT BELIEVE ME, PRESS"
430 PRINT"THAT LONG BAR, AT THE BOTTOM"
440 PRINT"OF THE KEYBOARD...OK ";N$;"?"
450 FOR I=1TO2000:NEXT
460 PRINT"EXCUSE ME WHILE I UNSCRAMBLE A"
470 PRINT"FEW FLIP-FLOPS....."
480 RANDOM
490 REM TABLES
500 TA=RND(50):IF TA<=15 THEN 500
510 REM START DISTANCE FROM ROOM
520 D=RND(150)
530 IF D<=50 THEN 520
540 REM CUSTOMERS IN BAR
550 P=RND(100)
560 IF P<=20 THEN 550
570 REM NUMBER OF WRONG DOORS
575 REM LEAVE TWO FOR FINAL
580 DR=RND(7)
590 IF DR<=4 THEN 580
600 REM READ WRONG DOOR TITLES
610 FOR I=1TODR:READ D$(I):U(I)=I:NEXT
615 FOR I=DR+1TO10:READ F$(I):NEXT
620 DATA WOMEN'S ROOM,ATTIC,EXIT,BASEMENT
630 DATA OFFICE,WINE CELLAR,LUNCH
640 DATA LAUNDRY,DRESSING,NECESSARY
650 J=DR-4:PQ=1:PJ=1:REM TIME / MEANIES
```

```
660 FOR I=1TO5:READ M$(I):M(I)=I
665 IF LEFT$(M$(I),1)<>"M" THEN 660
    ELSE NEXT
670 DATA MR. MUZZLE,MR. SMASH
680 DATA MS. LEAN TOO,MS. FOUL MOUTH
690 DATA MR. P. WADDLE
700 REM SOME STRING CONTENTS
710 K$="STILL":B$="MY CPU HURTS"
720 L$="LEFT":R$="RIGHT":Z$="ZZZZZZZZZZ"
730 FOR I=1TO1500:NEXT
740 PRINT
750 PRINT N$;" WE ARE NOW AT TABLE";TA
760 PRINT"MEANING, WE MUST TRAVEL";TA
770 PRINT"TABLES TO REACH THE BAR."
780 PRINT"TOTAL DISTANCE, I WOULD ESTIMATE"
790 PRINT"TO BE AROUND";D;"FEET."
800 PRINT"GLANCING AROUND(SOMETHING YOU'RE"
810 PRINT"NOT TO GOOD AT) I JUDGE.THERE
    ARE ABOUT";P
820 PRINT"CUSTOMERS WE MUST GET PAST..."
830 REM MESSAGE ---- B$
840 L=-64:I=0
850 L=L+64:I=I+1:FOR II=1TO500:NEXT
860 PRINT@L,STRING$(35,32)
870 PRINT@L+30,MID$(B$,I,1):IF L<=896 THEN
    850
880 CLS
890 GOSUB 1070:GOTO 920
900 PRINT"OK, LUSH...I MEAN ";N$
910 T=0:RETURN
920 GOSUB 900:GOSUB 925:GOTO 960
925 REM MEANIE SELECT
930 M=RND(5):IF M(M)=0 THEN 930
940 M$=M$(M):M(M)=0
950 IF LEFT$(M$,2)="MS" THEN C$=". SHE"
    ELSE C$=". HE"
955 RETURN
960 PRINT"GET UP AND MOVE ";T$;", AWAY"
970 PRINT"FROM ";M$;" ";C$;"'S A MEANIE,"
980 PRINT"WE DON'T WANT TO TANGLE WITH."
990 GOSUB 1000:GOTO 1130
1000 W$=INKEY$:T=T+1:IF T>=J*100 THEN 1025
1010 IF W$="" THEN 1000
1020 WW=ASC(W$):RETURN
1025 PRINT
1030 PRINT"PRESS THE ";R$;" ARROW, OR"
1040 PRINT"THE ";L$;" ARROW OR THE LONG"
1050 PRINT"BAR...YOU DRUNK, QUIT BOBBIN'"
```

```
1060 PRINT"YOUR HEAD !!":T=0:GOTO 990
1070 REM GET LEFT / RIGHT
1080 FOR I=1TO10
1090 Y=RND(2)
1100 IF Y=1 THEN T$=L$:GOTO 1120
1110 T$=R$
1120 NEXT:RETURN
1130 IF WW+85=93 THEN 1170
1140 IF WW+85=94 THEN 1170
1150 IF WW=32 THEN 1170
1160 PRINT:GOTO 1030
1170 WW=ASC(W$)
1180 IF ABS(T-J))=100 GOSUB 1520
1190 IF T$="RIGHT" AND WW+85=94 THEN 1440
1200 IF T$="LEFT" AND WW+85=93 THEN 1440
1210 IF WW=32 THEN PRINT:GOTO 1700
1215 IF T$="RIGHT" THEN F$=L$ ELSE F$=R$
1216 PRINT
1220 PRINT"I SAID, GO ";T$;", YOU WENT ";F$
1230 PRINT N$;", YOU ARE A LITTLE MORE"
1240 PRINT"DRUNK THAN I FIRST THOUGHT !!"
1250 RE=RND(6):UU=RND(5):M$=M$(UU)
1260 ON RE GOTO 1270,1290,1320,1345,1365,
     1400
1270 PRINT M$;" JUST BROKE PART"
1275 PRINT"OF YOUR FACE !!"
1280 GOTO 1430
1290 PRINT"YOU JUST STEPPED ON ";M$;" TOES,"
1300 PRINT"CAN'T YOU LISTEN TO DIRECTIONS
     ??"
1310 GOTO 1430
1320 PRINT"HOW DID THAT BOTTLE THAT ";M$
1330 PRINT"JUST BROKE OVER YOUR HEAD FEEL??"
1340 GOTO 1430
1345 IF JA=1 THEN 1250
1350 PRINT"MOST LUSHES LISTEN, YOU ARE AN"
1360 PRINT"EXCEPTION...DINGY !!!":GOTO 1430
1365 IF JA=1 THEN 1250
1370 PRINT"THREE KEYS TO PRESS, AND YOU MY"
1380 PRINT"DRUNK FRIEND, PRESS THE WRONG
     ONE !!"
1390 GOTO 1430
1400 PRINT"OF ALL THE MEANIES TO RUN IN
     TO !!!"
1410 PRINT"NEXT TIME ";M$;" WILL PROBABLY"
1420 PRINT"FINISH BREAKING YOUR FACE..."
1430 PRINT:E$="STANDS":GOTO 1660
1440 REM MOVE CORRECT ?
```

```
1450 PL=RND(10)
1460 IF PL<=DR THEN 1610 ELSE PRINT
1470 PRINT"I'M SORRY, MY MISTAKE, I TOLD"
1480 PRINT"YOU THE WRONG DIRECTION ";N$;"."
1490 PRINT"DINGY !! YOU SHOULDN'T HAVE"
1500 PRINT"LISTENED THIS TIME !!!"
1510 PRINT:JA=1:GOTO 1250
1520 REM SNORE MESSAGE
1530 REM TAKING TOO LONG / WRONG KEY
1540 FOR I=1TO LEN(Z$)
1550 PRINT MID$(Z$,I,1);
1560 FOR ER=1TO50:NEXT ER
1570 NEXT I:PRINT
1580 PRINT"THOUGHT I WOULD CUT A FEW"
1590 PRINT"Z'S WHILE YOU WERE LOOKING"
1600 PRINT"FOR THE KEYS.....":PRINT
1605 FOR I=1TO1200:NEXT:RETURN
1610 IF D<=50 THEN 1830 ELSE PRINT
1615 PRINT"WE'RE GETTING CLOSER,  ";N$
1620 PRINT"HOLD THAT BLADDER !!!":PRINT
1630 D=(D-INT(D/TA)-10):E$="IS NOW"
1640 IF D<=0 THEN 2500
1650 TA=TA-1:P=P-RND(5)
1660 PRINT"OUR DISTANCE ";E$;" AT";D;"FEET."
1665 IF TA<=0 PRINT"AND ";:GOTO 1675
1670 PRINT"WITH ";TA;"TABLES, AND"
1675 IF P<=0 THEN PRINT"NO CUSTOMERS TO GET
     BY !!":GOTO 1690
1676 IF P<=10 THEN K$="ONLY"
1680 PRINT K$;" ";P;"CUSTOMERS TO GET BY."
1685 IF RND(TA)<=10 AND XX=0 THEN 2300
1690 XX=0:GOTO 2100
1700 REM SPACE BAR ?
1710 PL=RND(5)
1720 IF PL<=2 THEN 1790
1730 PRINT"WHY? WHY DID YOU PRESS THE SPACE"
1740 PRINT"BAR?? OF ALL THE TIMES NOT TO"
1750 PRINT"BELIEVE ME, THIS TIME I WAS "
     ;R$;"."
1760 PRINT"YOU ARE AT THE SAME PLACE YOU"
1770 PRINT"WERE YOU...YOU DINGY #!/#&'!"
1780 PRINT:GOTO 1250
1790 PRINT"YOU WERE RIGHT, ";N$;" I TOLD"
1800 PRINT"A BLOODY LIE AND YOU"
1810 PRINT"DIDN'T FALL FOR IT !!!"
1820 PRINT:GOTO 1610
1830 REM WRONG DOOR MESSAGE
1840 IF PQ>DR THEN 1615
```

```
1850 DA=RND(DR)
1860 IF U(DA)=0 THEN 1850
1870 V$=D$(DA)
1880 U(DA)=0:PQ=PQ+1
1885 IF MID$(V$,2,1)="X" THEN 2040
1890 DA=RND(6):IF DA=6 THEN 1615
1900 ON DA GOTO 1910,1940,1970,2010,2040
1910 PRINT"YOU'VE JUST WALKED INTO THE ";V$
1920 PRINT"FEEL EMBARRASSED, YOU DINGY !!
1930 GOTO 2070
1940 PRINT"NO PLACE TO DRAIN A BLADDER HERE"
1945 IF LEFT$(V$,2)="OF" PRINT"IN THE ";V$;"
     ":GOTO 2070
1950 PRINT"THIS IS THE ";V$;
1955 IF RIGHT$(V$,2)="CH" PRINT"ROOM !!"
1960 PRINT:GOTO 2070
1970 PRINT"FEEL ODD ? YOU'VE JUST STEPPED"
1980 PRINT"(ALMOST) STRAIGHT INTO THE ";V$
1990 PRINT"HAVING PROBLEMS READING SIGNS ??"
2000 GOTO 2070
2010 PRINT"W R O N G  D O O R  D I N G Y
     ! !"
2020 PRINT"THIS IS THE ";V$;
2025 IF RIGHT$(V$,2)="CH" PRINT" ROOM"
2030 PRINT:GOTO 2070
2040 PRINT"GUESS WHAT YOU'VE JUST WALKED"
2045 IF LEFT$(V$,2)="EX" THEN 2065
2050 PRINT"INTO? YES, THE ";V$;
2055 IF RIGHT$(V$,2)="CH" PRINT"ROOM !!"
2060 PRINT:GOTO 1920
2065 PRINT"OUT THE ";V$;".":GOTO 2070
2070 PRINT:IF LEFT$(V$,2)="EX" THEN 2250
2080 GOTO 2410
2100 REM CONTINUE MOVEMENT TO HEAD
2110 PJ=PJ+1:GOSUB 900
2120 IF PJ<=5 GOSUB 930
2130 FOR I=1TO3500:NEXT:GOSUB 1070
2140 WE=RND(5):REM MOVEMENT MESSAGE SELECT
2150 ON WE GOTO 2160,2170,2180,2190,2200
2160 PRINT"WE WILL NOW MOVE ";T$;
2165 GOTO 2220
2170 PRINT"ALL DIRECTIONS POINT ";T$;
2175 GOTO 2220
2180 PRINT T$;" IS THE RIGHT DIRECTION ";
2185 GOTO 2220
2190 PRINT"LOOK ";T$;" THEN MOVE ";T$;
2195 GOTO 2220
2200 PRINT"DON'T STUMBLE NOW, DINGY !!!"
```

```
2210 PRINT"MOVE SLOWLY ";T$;
2220 IF PJ<=4 PRINT", AWAY":GOTO 970
2230 PRINT
2240 GOTO 990
2250 PRINT"GOING OUT THE ";V$;" WASN'T TOO"
2260 PRINT"SMOOTH...TURN AROUND, THEN WE'LL"
2270 PRINT"CONTINUE TO LOCATE THAT ROOM."
2280 FOR I=1TO3000:NEXT
2290 GOTO 2100
2300 REM REMOVE MORE CUSTOMERS
2305 IF P<=3 THEN XX=1:GOTO 1690
2310 AA=RND(5):DF=DF+1:XX=1
2315 IF DF>=4 THEN 2340 ELSE P=P-AA
2320 PRINT"EXCUSE ME, THAT'S";P;"CUSTOMERS."
2325 IF DF>=2 THEN MM$="MORE"
2330 PRINT"A FEW ";MM$;" JUST PASSED OUT..."
2340 IF RND(P)<=10 THEN 2360
2350 GOTO 2410
2360 FOR I=1TO2000:NEXT:IF DF>=7 THEN 24r0
2370 PRINT"HOLD IT !!! THE BOUNCER JUST"
2380 AA=RND(3):P=P-AA
2390 PRINT"CARRIED OUT";AA;"MORE, FOR"
2400 PRINT"FIGHTING DURING HAPPY HOUR..."
2410 FOR I=1TO2000:NEXT
2420 PRINT
2430 E$="STANDS":JA=0
2440 GOTO 1660
2500 REM FINAL -- ROOM REACHED
2510 PRINT"YOU'VE REACHED THE ROOM !!!"
2520 FOR I=1TO2500:NEXT
2530 I=RND(10)
2540 IF I<=DR THEN 2530
2550 PRINT"IF YOU CAN KEEP YOUR EYES OPEN"
2560 PRINT"LONG ENOUGH...RAISE YOUR HEAD"
2570 PRINT"AND READ THE SIGN ON THE DOOR."
2580 D$=F$(I)
2590 X=0
2600 FOR I=1TO900:NEXT
2610 PRINT:X=X+1
2620 IF X=7 PRINT@985,D$;" ROOM"
2630 IF X<=14 THEN 2600
2640 IF LEFT$(D$,1)="N" THEN 2690
2650 PRINT
2660 PRINT"** T H E  W R O N G  D O O R **"
2670 PRINT"ALL THIS FOR NOTHING, DINGY !!!"
2680 GOTO 2760
2690 PRINT"THIS IS IT !!!"
2700 PRINT"THE RIGHT NECESSARY ROOM !!!"
```

```
2710 PRINT"RELIEF IS ON THE WAY..."
2720 PRINT
2730 FOR I=1TO4000:NEXT
2740 PRINT"NOW THAT YOU ARE FEELING MUCH"
2750 PRINT"BETTER, SHALL WE HAVE....."
2760 PRINT"ANOTHER RUN OF NECESSARY ROOM";
2770 INPUT X$
2780 IF X$<>"Y" AND X$<>"YES" THEN 2810
2790 PRINT"GET TO DRINKING..HERE WE GO !!"
2800 FOR I=1TO1200:NEXT:CLEAR:DD=1:GOTO 275
2810 PRINT
2820 PRINT"END"
2830 END
```

Variables and Strings

DD - Instructions not needed (when equal to 1)
N$ - User's name
TA - Number of tables
D - Distance to room
P - Customers in bar
DR - Wrong doors, D$(I)
M$(I) - Meanies in bar
J - Time loop
PJ - Moves toward room
M - Selection of meanie
WW - User's key closure
RE - For random message, along with UU and M$
PL - Correct move (random)

Explanation of the Program Lines

Lines 10-270 display the general instructions if they are required.

Line 280 inputs the user's name.

Lines 290-470 display the rest of the instructions.

Lines 480-720 set the variables for play and read all the data statements.

Lines 750-820 are messages to the user concerning distances, tables, and customers.

Lines 830-870 let the user know how the computer feels.

Lines 900-910 display a message used by GOSUB statements throughout the program.

Lines 925-955 select a meanie.

Lines 960-980 tell the user which direction to take.

Lines 1000-1020 form the INKEY$ routine for the user's direction entry.

Lines 1025-1060 display a message indicating that the user is taking too long to decide.

Lines 1070-1120 get a left/right direction for the user.

Lines 1130-1160 sort the key closures indicating the direction of movement.

Lines 1170-1430 display more messages concerning movement to the room.

Lines 1440-1510 display a wrong move message.

Lines 1520-1605 display a message indicating that the user is taking too long.

Lines 1610-1690 display a message indicating that the distance and number of tables are decreasing.

Lines 1700-1820 display messages in response to the pressing of the space bar.

Lines 1830-2440 display various messages concerning the user's position.

Lines 2500-2830 display the final message printed on door of the room the user has come to.

TROUBLES

Depending upon the nature of your particular troubles, the computer will try and give you a solution. Of course, it would take more memory space that any computer has to cover every aspect of a person's troubles, so be patient with your computer. Give it the precise information it requires and who knows?

Sample Run

```
        TROUBLES

SO YOU HAVE SOME TROUBLES AND
YOU ARE TURNING TO ME, YOUR COMPUTER,
I WILL HELP YOU IN EVERY WAY I
CAN, USING LOGICAL DEDUCTIONS.
PLEASE BE SPECIFIC
WHEN ANSWERING ALL QUESTIONS
I WILL SUBMIT TO YOU. THE MORE
SPECIFIC YOU ARE, THE BETTER
YOUR ANSWER WILL BE.
USE NO COMMAS, SEMICOLONS,
COLONS OR PERIODS.

YOUR FIRST & LAST NAME?TARGET MOORE
MALE (M) OR FEMALE (F)?
TRY AGAIN: CHECK IF NECESSARY.
```

MALE (M) OR FEMALE (F)?M
TROUBLES

DATA COLLECTED THUS FAR:
NAME TARGET MOORE
SEX: MALE
CONFIRM (Y/N)?Y
WITH MORE THAN 1 WORD AND LESS
THAN 10, DESCRIBE YOUR TROUBLES? WOMEN
BE A LITTLE MORE SPECIFIC, WHAT
ABOUT WOMEN?
WITH MORE THAN 1 WORD AND LESS
THAN 10 DESCRIBE YOUR TROUBLES?
CAN'T GET ENOUGH RESPONSE FROM WOMEN
TROUBLES

SORTING WORDS

JUST HOW MANY WOMEN?3

GOOD GRIEF !!
IS ANY ONE OF THESE YOUR WIFE?YES

DO YOU FEEL YOU ARE CHEATING ON
YOUR WIFE AND YOURSELF?YES

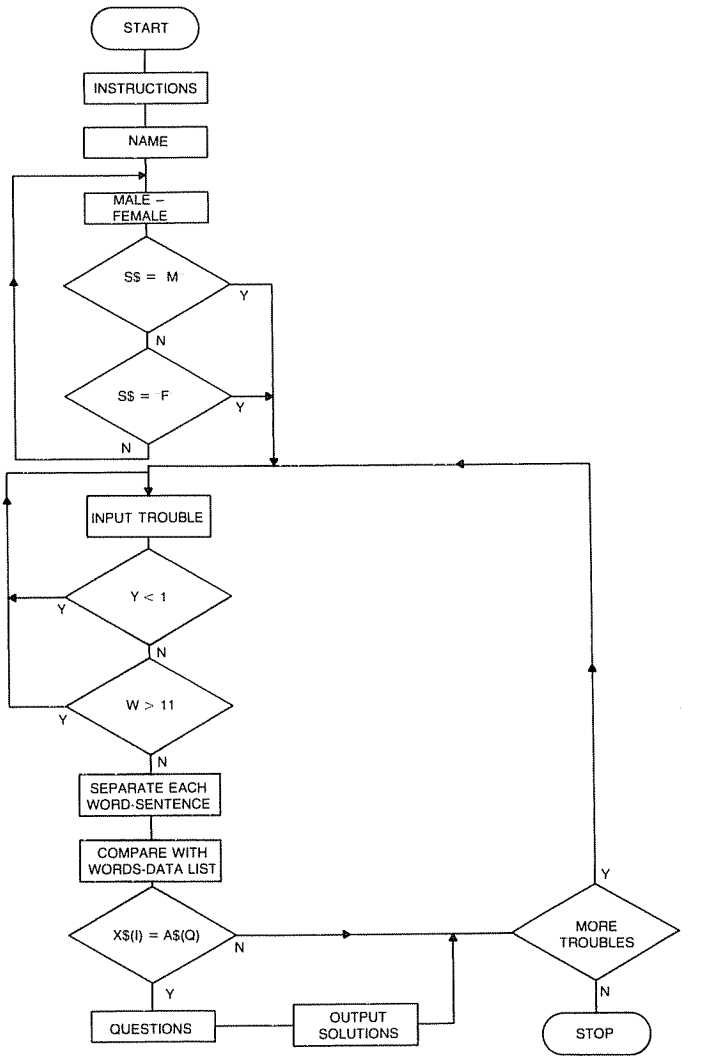
YOU SHOULD FORGET THE OTHER
WOMEN BEFORE YOU GET DEEPER THAN
WHAT YOU ARE NOW...TARGET

DO YOU AGREE

? NO

WHY DON'T YOU AGREE TARGET ?I CHOOSE TO PLAY
THE FIELD

BECAUSE YOU CHOOSE TO PLAY THE FIELD?
HOW LONG HAS THIS BEEN
GOING ON TARGET
SELECT ONE:
1> DAYS



Flowchart for *Troubles*.

2> WEEKS
3> MONTHS
4> YEARS
?4

DID THESE WOMEN FIND OUT
ABOUT ONE ANOTHER TARGET ?YES

THAT'S THE TROUBLE: JUST STICK
TO ONE WOMAN FOR AWHILE
THINGS WILL WORK OUT MUCH
BETTER. YOU'LL SEE, TARGET .
LIKE TO TRY ANOTHER?NO

END OF PROGRAM...

Program Listing

```
 E10 REM PROGRAM TITLE: TROUBLES
 20 CLS:CLEAR 1200:GOSUB 30:GOTO 50
 30 PRINT TAB(10)"T R O U B L E S"
 40 PRINT:RETURN
 50 PRINT"SO YOU HAVE SOME TROUBLES AND"
 60 PRINT"YOU ARE TURNING TO ME, YOUR
    COMPUTER."
 70 PRINT"I WILL HELP YOU IN EVERY WAY I"
 80 PRINT"CAN USING LOGICAL DEDUCTIONS."
 90 PRINT"PLEASE BE SPECIFIC"
100 PRINT"WHEN ANSWERING ALL QUESTIONS"
110 PRINT"I WILL SUBMIT TO YOU. THE MORE"
120 PRINT"SPECIFIC YOU ARE, THE BETTER"
130 PRINT"YOUR ANSWER WILL BE."
140 PRINT"USE NO COMMAS, SEMICOLONS,"
145 PRINT"COLONS OR PERIODS."
150 PRINT:PRINT"YOUR FIRST & LAST NAME";
160 INPUT N$:GOSUB 490
170 INPUT"MALE (M) OR FEMALE (F)";S$
180 IF S$="M" THEN S$="MALE":GOTO 220
190 IF S$="F" THEN S$="FEMALE":GOTO 220
200 PRINT"TRY AGAIN; CHECK IF NECESSARY."
210 GOTO 170
220 CLS
230 GOSUB 30
240 REM BEGIN
250 PRINT"DATA COLLECTED THUS FAR:"
260 PRINT"NAME:     ";N$
```

```
270 PRINT"SEX:      ";S$
290 GOSUB 550
300 INPUT"CONFIRM (Y/N)";A$
310 IF A$<>"Y" THEN F$="":GOTO 140
320 DIM X$(46),A(12),A$(12)
325 P$="":W=1:PRINT
330 PRINT"WITH MORE THAN 1 WORD AND LESS"
340 PRINT"THAN 10, DESCRIBE YOUR TROUBLES"
345 IF Y=1 THEN Y=0:PRINT"NO PERIODS..."
350 INPUT P$
360 FOR I=1 TO LEN(P$)
370 IF MID$(P$,I,1)=" " THEN 390
375 IF MID$(P$,I,1)="." THEN Y=1:GOTO 325
380 NEXT:IF W<=1 THEN 455 ELSE GOTO 470
390 REM WORD COUNT
400 W=W+1
410 IF W>=11 THEN 430
420 GOTO 380
430 IF LEN(P$)<=200 THEN 470
440 PRINT"YOU'VE ENTERED";W;"WORDS ";N$
445 PRINT"REENTER A SHORTER VERSION."
450 GOTO 325
455 PRINT"BE A LITTLE MORE SPECIFIC, WHAT"
460 PRINT"ABOUT ";P$;"?"
465 GOTO 325
470 REM DEFINE PARTS OF PROBLEM
480 CLS:GOSUB 30:GOTO 580
490 REM FIRST NAME BASIS
500 FOR I=1 TO LEN(N$)
510 F$=F$+MID$(N$,I,1)
520 IF MID$(N$,I,1)=" " THEN RETURN
530 NEXT
540 RETURN
550 REM TIME LOOP
560 FOR TI=1TO1000:NEXT
570 RETURN
580 REM REPLACEMENT WORDS / WORD DATA
590 FOR I=1 TO 46:READ X$(I):NEXT:L=1:Q=1
600 REM WORD COUNT / WORD IN DATA LIST
610 I=1:K=LEN(P$)
620 IF MID$(P$,I,1)=" " THEN 650
630 I=I+1:IF I=K THEN 670
640 GOTO 620
650 A(Q)=L:L=I:Q=Q+1:REM WORD SEPARATION
660 GOTO 630
670 A(Q)=L
680 A(1)=0:I=1:Z=Q:Q=1
690 FOR L=A(I)+1 TO A(I+1)-1
```

```
 700 A$(Q)=A$(Q)+MID$(P$,L,1)
 710 NEXT:Q=Q+1
 720 IF I=Z THEN 740
 730 I=I+1:GOTO 690
 740 REM LAST WORD IN SENTENCE
 750 FOR L=A(I)+1 TO LEN(P$)
 760 A$(Q)=A$(Q)+MID$(P$,L,1)
 770 NEXT:Q=1
 780 A$(Z)=A$(Z+1):A$(Z+1)=""
 790 REM COMPARE EACH WORD IN SENTENCE
 800 REM WITH WORDS IN DATA LIST FOR MATCH
 810 I=1
 820 IF X$(I)=A$(Q) THEN 1010
 830 I=I+1:IF I>=30 AND Q>=3 THEN 890
 840 IF I>=47 THEN 860
 850 GOTO 820
 860 Q=Q+1
 870 IF Q>=Z+1 THEN 910
 880 GOTO 810
 890 PRINT@75,"SORTING WORDS"
 900 GOTO 840
 910 REM WORD NOT IN FILE
 920 PRINT
 930 PRINT"I AM REALLY SORRY, BUT I
     AM UNABLE"
 940 PRINT"TO DISCUSS THIS PARTICULAR
     PROBLEM"
 950 PRINT"WITH YOU. PERHAPS YOU WOULD"
 960 PRINT"LIKE TO TRY ANOTHER";
 970 GOSUB 980:GOTO 990
 980 INPUT A$:PRINT:RETURN:REM STRING ENTRY
 990 IF A$="Y" OR A$="YES" THEN 1180
1000 GOTO 4620
1010 REM WORD LOCATED / START OUTPUT
1020 W$=X$(I)
1030 M=I
1040 IF M>=1 AND M<=4 THEN 1190
1050 IF M>=5 AND M<=8 THEN 1190
1060 IF M>=9 AND M<=16 THEN 2700
1070 IF M>=20 AND M<=28 THEN 3300
1080 IF M>=29 AND M<=31 THEN 2000
1090 IF M>=32 AND M<=35 THEN 3600
1100 IF M>=36 AND M<=42 THEN 4030
1110 IF M=43 OR M=45 THEN 2100
1120 IF M=44 OR M=46 THEN 2100
1180 RESTORE:GOTO 325
1190 IF M=2 OR M=4 OR M=6 OR M=7 THEN 2100
1200 PRINT"JUST HOW MANY ";W$;
```

```
1205 IF M=5 OR M=8 THEN 5820
1210 GOSUB 980:A=VAL(A$):R$=X$(43)
1220 AA=A:IF AA>=3 PRINT"GOOD GRIEF !!"
1225 IF A=1 THEN 1270
1230 PRINT"ARE ANY ONE OF THESE YOUR ";R$;
1240 GOSUB 980
1250 IF A$="YES" OR A$="Y" THEN V=1
1260 IF V=1 THEN 1390
1270 PRINT"SO YOU'RE JUST PLAYING THE FIELD."
1280 PRINT"ARE YOU SEARCHING FOR A ";R$;
1290 GOSUB 980
1300 IF A$="YES" OR A$="Y" THEN 1750
1310 PRINT"YOU PLAN ON STAYING SINGLE";
1320 GOSUB 980
1330 IF A$="YES" OR A$="Y" THEN 1750
1340 PRINT"YOU'RE NOT GOING TO STAY SINGLE,"
1350 PRINT"BUT AT THE SAME TIME YOU'RE"
1360 PRINT"LOOKING FOR A ";R$;" YOU"
1370 PRINT"HAVE GOT TROUBLES !!"
1380 GOTO 4600
1390 PRINT"DO YOU FEEL YOU ARE CHEATING ON"
1400 PRINT"YOUR ";R$;" AND YOURSELF";
1410 GOSUB 980
1420 IF A$="YES" OR A$="Y" THEN 1560
1430 PRINT"YOU DON'T ";F$;
1440 GOSUB 980:IF A$="YES" THEN 1560
1450 PRINT"ONE OF THOSE HARD-NOSED HUMANS,"
1460 PRINT"HUH? I SUGGEST YOU NIP IT"
1470 PRINT"AT THE BUD, EXPLAIN THINGS TO"
1480 PRINT"YOUR ";R$;" AND FORGET THE OTHER"
1490 PRINT W$;" BEFORE YOU GET DEEPER THAN"
1500 PRINT"YOU ARE NOW...";F$
1505 PRINT
1510 PRINT"DO YOU AGREE";:GOSUB 980
1515 IF KL=1 THEN KL=0:GOTO 4700
1520 IF A$="NO" OR A$="N" THEN 1580
1530 PRINT"I THOUGHT YOU'D SEE THINGS"
1540 PRINT"MY WAY ";F$
1550 GOTO 4600
1560 PRINT"YOU SHOULD FORGET THE OTHER"
1570 GOTO 1490
1580 PRINT"WHY DON'T YOU AGREE ";F$;
1590 GOSUB 980:GOSUB 5740
1600 PRINT"BECAUSE ";A$;"?"
1610 PRINT"HOW LONG HAS THIS BEEN"
1620 PRINT"GOING ON ";F$
1630 PRINT"SELECT ONE:"
1640 PRINT"1> DAYS"
```

```
1650 PRINT"2) WEEKS"
1660 PRINT"3) MONTHS"
1670 PRINT"4) YEARS"
1680 GOSUB 980:A=VAL(A$)
1690 IF A=1 THEN DA$="DAYS":GOTO 1740
1700 IF A=2 THEN DA$="WEEKS":GOTO 1740
1710 IF A=3 THEN DA$="MONTHS":GOTO 1740
1720 IF A=4 THEN DA$="YEARS":GOTO 1740
1730 PRINT:GOTO 1610
1740 IF JK=1 THEN JK=0:RETURN
1750 IF KJ=1 THEN KJ=0:PRINT:GOTO 3640
1755 PRINT"DID THESE ";W$;" FIND OUT"
1760 PRINT"ABOUT ONE ANOTHER ";F$;
1770 GOSUB 980
1780 IF A$="YES" OR A$="Y" THEN 1800
1790 GOTO 1850
1800 PRINT"THAT'S THE TROUBLE! JUST STICK"
1810 PRINT"TO ONE ";X$(4);" FOR A WHILE."
1820 PRINT"THINGS WILL WORK OUT MUCH"
1830 PRINT"BETTER, YOU'LL SEE ";F$;"."
1840 GOTO 4600
1850 PRINT"THEY DIDN'T? SO THE TROUBLE"
1860 PRINT"MUST BE WITH YOU. WHY IS IT"
1870 PRINT"NECESSARY FOR YOU TO SEE";AA
1880 PRINT W$;" AT ONCE? HAVE YOU TRIED"
1890 PRINT"JUST ONE";:GOSUB 980
1900 IF A$="YES" OR A$="Y" THEN 1930
1910 PRINT"YOU HAVEN'T? TRY IT!"
1920 GOTO 1820
1930 PRINT"YOU HAVE ? I ASSUME THAT YOU"
1940 PRINT"DIDN'T LIKE IT ";F$;
1950 GOSUB 980
1960 PRINT"I MIGHT SUGGEST YOU KEEP TO"
1970 Z=RND(6):IF Z=1 THEN 1970
1980 PRINT"YOURSELF FOR AT LEAST";Z;"WEEKS."
1990 GOTO 1820
2000 PRINT"ARE YOU AWARE WHAT ";X$(30)
2010 PRINT"WILL DO TO YOUR HEALTH";:GOSUB 980
2020 IF A$="N" OR A$="NO" THEN 2060
2030 PRINT"YOU ARE? THEN YOU SHOULD TAKE"
2040 PRINT"STEPS TO STOP...NOW ";F$;"!!"
2050 GOTO 4600
2060 PRINT"THROAT CANCER, LUNG CANCER, A"
2070 PRINT"SHORTER LIFE EXPECTANCY. JUST"
2080 PRINT"TO NAME A FEW. TAKE A FEW"
2090 GOTO 2040
2100 IF M<=4 OR M=44 OR M=43 THEN R$=X$(4)
     :JJ=1:GOTO 2120
```

```
2110 R$=X$(6)
2120 PRINT"WHAT IS THE SINGLE MOST WORD"
2130 PRINT"THAT WOULD BEST DESCRIBE YOUR"
2140 PRINT"TROUBLES WITH THIS ";R$;
2150 GOSUB 980
2160 E$=A$
2220 PRINT"NOT ENOUGH ";E$;" ";F$;
2230 GOSUB 980
2235 IF A$<>"YES" AND A$<>"RIGHT" THEN 2280
2240 PRINT"SO YOU ARE JUST NOT SATISFIED"
2250 PRINT"WITH THE WAY THINGS ARE GOING"
2260 PRINT"WITH YOUR ";E$;
2270 GOSUB 980
2280 PRINT"OH I SEE ";F$;". HAVE YOU"
2290 PRINT"DISCUSSED THIS WITH ";
2300 IF JJ=1 PRINT"HER";:GOTO 2320
2310 PRINT"HIM";
2320 GOSUB 980
2330 IF A$="N" OR A$="NO" THEN 2360
2340 PRINT"AND I ASSUME IT DIDN'T HELP."
2350 JK=1:GOSUB550:PRINT:GOSUB 1610:GOTO
     2430
2360 PRINT"TRY IT; THINGS JUST MIGHT"
2370 PRINT"WORK OUT FOR THE BEST. IF"
2380 PRINT"NOT, CONSULT WITH ME AGAIN"
2390 PRINT"AND I WILL GIVE YOU A"
2400 PRINT"DIFFERENT ANSWER TO YOUR"
2410 PRINT"TROUBLES, ";F$;"."
2420 KL=1:GOTO 1505
2430 PRINT"DURING THE PAST ";DA$;" HOW"
2440 PRINT"MANY TIMES HAVE YOU DISCUSSED"
2450 PRINT"THIS WITH ";
2460 IF JJ=1 PRINT"HER";:GOTO 2480
2470 PRINT"HIM";
2480 GOSUB 980
2490 A=VAL(A$)
2500 PRINT"DID ANY OF THOSE";A;"TIMES HELP"
2510 PRINT"AT ALL ";F$;
2520 GOSUB 980
2530 IF A$="Y" OR A$="YES" THEN 2580
2535 IF A<=4 THEN 2580
2540 PRINT"I WILL HAVE TO TELL YOU TO"
2550 PRINT"SEEK SOME OUTSIDE HELP ON"
2560 PRINT"THIS MATTER...";F$:KL=1
2570 X=RND(2):IF X=1 GOTO 4600 ELSE 1505
2580 PRINT"CONTINUE THEN TO DISCUSS THIS"
2590 PRINT"PROBLEM OF ";E$;" WITH ";
2600 IF JJ=1 PRINT"HER":GOTO 2620
```

```
2610 PRINT"HIM"
2620 PRINT"IF YOU'VE PROGRESSED THIS"
2630 PRINT"FAR, THINGS WILL SURELY GET"
2640 PRINT"BETTER...";F$
2650 GOTO 4600
2700 REM MONEY
2710 PRINT"DO YOU HAVE A STEADY ";
2720 PRINT X$(36);" ";F$;
2730 GOSUB 980
2740 IF A$="N" OR A$="NO" THEN 2880
2750 IF W$()X$(13) THEN 2770
2760 GOTO 2820
2770 PRINT"BUT YOU ARE IN ";X$(13);:
     GOSUB 980
2780 IF A$="N" OR A$="NO" THEN 2990
2790 PRINT"HAVE YOU CONSIDERED A ";X$(14);
2800 GOSUB 980
2810 IF A$="Y" OR A$="YES" THEN 3020
2820 PRINT"FINANCE CHARGES ARE HIGH, BUT"
2830 PRINT"ONE PAYMENT WOULD PROBABLY BE"
2840 PRINT"EASIER THAN SEVERAL TO DIFFERENT"
2850 PRINT"ONES ";F$;". YOU SHOULD"
2860 PRINT"CONSIDER A ";X$(14);"."
2870 GOTO 4600
2880 PRINT"ARE YOU WORKING PART-TIME";
2890 GOSUB 980
2900 IF A$="N" OR A$="NO" THEN 3120
2910 PRINT"O.K. ";F$;"YOU HAVE GOT A";X$(36)
2920 PRINT"BUT, YOU ARE ONLY WORKING"
2930 PRINT"PART-TIME. ARE YOU DEEPLY IN ";
2940 PRINT X$(13);:GOSUB 980
2950 IF A$="Y" OR A$="YES" THEN 3090
2960 PRINT"SO I WILL ASSUME THAT YOU JUST"
2970 PRINT"NEED SOME EXTRA CASH..."
2980 GOTO 2790
2990 PRINT"YOU HAVE A STEADY ";X$(36);" YOU"
3000 PRINT"ARE NOT IN ";X$(13);"....."
3010 GOTO 2960
3020 PRINT"DID THE ";X$(41);" TURN YOU DOWN";
3030 GOSUB 980
3040 IF A$="Y" OR A$="YES" THEN 3250
3050 PRINT"YOU NEED ";X$(11);", YOU APPLIED"
3060 PRINT"FOR A ";X$(14);", THEY DIDN'T"
3070 PRINT"TURN YOU DOWN. WHAT'S THE BEEF";
3080 GOSUB 980:JJ=1:GOTO 3250
3090 PRINT"SO THAT'S IT !!"
3100 PRINT"SEEK A HIGHER PAYING JOB ";F$;"."
3110 GOTO 4600
```

```
3120 PRINT"ARE YOU DISABLED ";F$;:GOSUB 980
3130 IF A$="N" OR A$="NO" THEN 3200
3140 PRINT"THAT EXPLAINS THE ";X$(11);
3150 PRINT" TROUBLE."
3160 PRINT"I MIGHT SUGGEST YOU SEARCH FOR A"
3170 PRINT"NICE FINANCE ";X$(41);" THAT WILL"
3180 PRINT"UNDERSTAND YOUR PROBLEM. TODAYS"
3190 GOTO 2820
3200 PRINT"I WILL HAVE TO ASSUME FROM ALL"
3210 PRINT"OF YOU ANSWERS THAT YOU ARE JUST"
3220 PRINT"A LAZY PERSON ";F$;" LOOKING"
3230 PRINT"FOR EASY CASH, GET A ";X$(36);"!!"
3240 GOTO 4600
3250 PRINT"WAIT A FEW MONTHS, IF POSSIBLE"
3260 PRINT"AND REAPPLY FOR THE ";X$(14)
3265 IF JJ=1 THEN JJ=0:GOTO 3290
3270 PRINT"THEY SHOULDN'T TURN YOU DOWN A"
3280 PRINT"SECOND TIME ";F$;"."
3290 GOTO 4600
3300 IF M>=20 AND M<=22 THEN 2790
3305 IF M=25 OR M=26 THEN JX=1:GOTO 4270
3310 PRINT"HAVE YOU DISCUSSED THIS WITH"
3320 PRINT"THE ";X$(25);" OR THE ";X$(26);
3330 GOSUB 980
3340 IF A$="N" OR A$="NO" THEN 3520
3350 PRINT"DID THIS SOLVE ANYTHING ";F$;
3360 GOSUB 980
3370 IF A$="N" OR A$="NO" THEN 3400
3380 PRINT"IS THIS A DUPLEX";:GOSUB 980
3390 IF A$="N" OR A$="NO" THEN 3540
3400 PRINT"HAVE YOU TALKED TO OTHER TENANTS"
3410 PRINT"ABOUT YOUR SITUATION";:GOSUB 980
3420 IF A$="N" OR A$="NO" THEN 3560
3430 PRINT"AND THIS, EVIDENTLY, DIDN'T HELP."
3440 PRINT"IF YOU HAVE IN FACT TALKED TO"
3450 PRINT"THE ";X$(26);" ABOUT THIS"
3460 PRINT"AND NOTHING WAS SOLVED, I WOULD"
3470 PRINT"SUGGEST MAKING UP A SCHEME TO"
3480 PRINT"BREAK THE ";X$(27);" OR LEASE."
3490 PRINT"SEEK A DIFFERENT LOCATION WITH"
3500 PRINT"OTHER ";X$(24);" ";F$;"."
3510 GOTO 4600
3520 PRINT"DO SO NOW, THEN ANSWER THE NEXT"
3530 PRINT"SET OF QUESTIONS, ";F$:GOTO 3350
3540 PRINT"YOU LIVE IN A SINGLE ";X$(23)
3550 GOTO 3440
3560 PRINT"DO SO NOW, THEY MIGHT BE GOING"
3570 PRINT"THROUGH THE SAME THING. IF THIS"
```

```
3580 PRINT"FAILS, ";F$;" I WOULD"
3590 GOTO 3470
3600 PRINT"WHY DO YOU FEEL ";W$;" ";F$;
3610 GOSUB 980
3620 PRINT"I FELT THAT WAY ONCE, UNTIL I"
3630 PRINT"RECEIVED A NEW RAM."
3640 PRINT"DO YOU HAVE FRIENDS ";F$;
3650 GOSUB 980
3660 IF A$="N" OR A$="NO" THEN 3830
3670 PRINT"DISCUSSING DIFFERENT THINGS WITH"
3680 PRINT"THEM SHOULD MAKE YOU FEEL SOME-"
3690 PRINT"WHAT BETTER. HOW OFTEN HAVE YOU"
3700 PRINT"FELT THIS WAY ";F$:JK=1
3710 GOSUB 1630
3720 IF A=4 THEN 3800
3730 PRINT"SO IT HASN'T BEEN GOING ON TOO"
3740 PRINT"LONG ";F$;". WHAT IS THE ONE"
3750 PRINT" THING THAT WOULD CHEER YOU MOST"
3760 PRINT"AT THIS MOMENT";:GOSUB 980
3770 PRINT"THEN DO THIS NOW ";F$;". IF"
3780 PRINT"ALL FAILS YOU CAN RETURN TO ME."
3790 KL=1:GOTO 1505
3800 PRINT"THAT IS A LONG TIME TO BE IN THIS"
3810 PRINT"STATE OF MIND ";F$;". WHAT IS THE"
3820 GOTO 3750
3830 PRINT"DO YOU LIVE WHERE YOU COULD MEET"
3840 PRINT"SOME NEW FRIENDS ";F$;:GOSUB 980
3850 IF A$="N" OR A$="NO" THEN 3920
3860 PRINT"WHY DON'T YOU GET OUT AND MEET"
3870 PRINT"SOME NOW? IS SOMETHING ELSE"
3880 PRINT"HOLDING YOU BACK";:GOSUB 980
3890 PRINT"OH, I SEE ";F$:GOTO 3920
3900 IF S$="MALE" THEN R$=X$(43):GOTO 1280
3910 R$=X$(45):JJ=1:GOTO 1280
3920 PRINT"HOW ABOUT MOVING TO A"
3930 PRINT"DIFFERENT LOCATION ";F$;:GOSUB 980
3940 IF A$="N" OR A$="NO" THEN 3980
3950 PRINT"I BELIEVE THINGS WILL WORK OUT"
3960 PRINT"FOR THE BEST WITH THIS MOVE."
3970 GOTO 4600
3980 PRINT"ARE YOU TRYING TO BE CONTRARY ";F$;
3990 GOSUB 980
4000 PRINT"LET'S TRY A DIFFERENT APPROACH TO"
4010 PRINT"THIS SUBJECT ";F$:GOSUB 550
4015 IF M=43 OR M=45 THEN PRINT:GOTO 3640
4020 KJ=1:GOTO 3900
4030 IF M=37 OR M=38 THEN 4270
4040 PRINT"WHAT'S THE TROUBLE WITH YOUR ";W$;
```

```
4050 GOSUB 980:GOTO 4540
4060 IF W$=X$(36) THEN 4080
4070 IF M=41 OR M=42 THEN 4120
4080 PRINT"HAVE YOU ASKED FOR A RAISE IN PAY";
4090 GOSUB 980
4100 IF A$="N" OR A$="NO" THEN 4520
4110 PRINT"I ASSUME YOU WERE TURNED DOWN."
4120 PRINT"DO YOU BELONG TO A UNION ";F$;
4130 GOSUB 980
4140 IF A$="N" OR A$="NO" THEN 4340
4150 PRINT"DO OTHERS FEEL THE WAY YOU DO";
4160 GOSUB 980
4170 IF A$="N" OR A$="NO" THEN 4440
4180 PRINT"HAVE ALL OF YOU MET TO DISCUSS"
4190 PRINT"A POSSIBLE STRIKE ";F$;
4200 GOSUB 980
4210 IF A$="N" OR A$="NO" THEN 4440
4220 PRINT"AGAIN, I ASSUME THINGS DIDN'T"
4230 PRINT"TURN OUT WELL ";F$;". I WOULD"
4240 PRINT"SUGGEST THAT YOU TRY AND LOCATE"
4250 PRINT"ANOTHER ";X$(36);" ";F$;"."
4260 GOTO 4600
4270 PRINT"WHAT EXACTLY IS THE PROBLEM"
4280 PRINT"WITH YOUR ";W$;" ";F$;:GOSUB 980
4290 PRINT"HAVE YOU BROUGHT THIS TO HIS OR"
4300 PRINT"HER ATTENTION, ";F$;:GOSUB 980
4310 IF A$="N" OR A$="NO" THEN 4455
4320 PRINT"YOU HAVE AND YOU FEEL YOU AREN'T"
4330 PRINT"BEING TREATED FAIRLY";:GOSUB 980
4335 IF JX=1 THEN 3400
4340 PRINT"HOW LONG HAVE YOU BEEN EMPLOYED"
4350 PRINT"WITH THIS ";X$(41);" ";F$;
4360 JK=1:GOSUB 1630
4370 IF A=4 THEN 4390
4380 GOTO 3100
4390 PRINT"YOU'VE BEEN WITH THIS ";X$(41)
4400 PRINT"FOR QUITE AWHILE. IF POSSIBLE"
4410 PRINT"YOU SHOULD STICK WITH IT FOR AS"
4420 PRINT"LONG AS YOU CAN ";F$
4430 GOTO 4600
4440 PRINT"YOU ARE ALONE ON THIS MATTER."
4450 GOTO 3100
4455 IF JX=1 THEN 3400
4460 PRINT"SIT DOWN WITH YOUR ";X$(37);"
4470 PRINT"DISCUSS THIS MATTER FULLY WITH"
4480 PRINT"HIM OR HER.IF THEY DO NOT UNDER-"
4490 PRINT"STAND YOUR PROBLEM ";F$;" I"
4500 PRINT"WOULD SEEK ANOTHER JOB."
```

```
4510 PRINT:GOTO 4120
4520 PRINT"TRY IT, ";F$;"!!":GOTO 4120
4530 PRINT"I ";:GOTO 4240
4540 FOR U=1 TO LEN(A$)
4550 IF MID$(A$,U,4)=X$(37) THEN 4580
4560 IF MID$(A$,U,6)=X$(38) THEN 4590
4570 NEXT:GOTO 4060
4580 W$=MID$(A$,U,4):GOTO 4595
4590 W$=MID$(A$,U,6)
4595 GOTO 4270
4600 GOSUB 550:PRINT
4610 GOTO 960
4620 PRINT
4630 PRINT"END OF PROGRAM..."
4640 END
4700 IF A$="N" OR A$="NO" THEN 3980
4710 PRINT"YOU DO AGREE, I FIGURED YOU WOULD"
4720 PRINT F$;". DO YOU ALWAYS TEND TO
4730 PRINT"AGREE WITH COMPUTERS";
4740 GOSUB 980
4750 IF A$="N" OR A$="NO" THEN 3980
4760 PRINT"SO THEN YOU AGREE WITH COMPUTERS"
4770 PRINT"ALMOST CONSTANTLY ";F$;"."
4780 PRINT"WOULD YOU AGREE IF I SAID THAT"
4790 PRINT"IF YOU PRESS ANOTHER"
4800 PRINT"KEY, I COULD KILL THIS WHOLE"
4810 PRINT"PROGRAM, ";F$;:GOSUB 980
4820 IF A$="N" OR A$="NO" THEN CLS:GOTO 4850
4830 PRINT"GOOD THING YOU SAID ";A$;"..."
4840 GOTO 4600
4850 FOR TT=1 TO 10000:NEXT
4860 PRINT"WE'RE YOU SCARED....";F$;"?"
4870 GOTO 4600
5200 REM WORDS / DATA BANK
5210 DATA GIRLS,GIRL,WOMEN,WOMAN,MEN,MAN
5220 DATA BOY,BOYS,BILLS,BILL,MONEY
5230 DATA FINANCIAL,DEBT,LOAN,TAX,TAXES
5240 DATA CAR,CARS,AUTO,HOUSE,HOUSES
5250 DATA HOME,APARTMENT,APARTMENTS
5260 DATA LANDLORD,OWNER,CONTRACT,RENT
5270 DATA CIGARETTES,SMOKING,CIGARETTE
5280 DATA LONELY,ALONE,DEPRESSED,AFFRAID
5290 DATA JOB,BOSS,FOREMAN,SALARY,INCOME
5300 DATA COMPANY,INDUSTRY,WIFE
5310 DATA GIRLFRIEND,HUSBAND,BOYFRIEND
5740 REM SHORTEN STRING
5750 U$=A$:A$=""
5770 FOR I=1 TO LEN(U$)
```

```
5780 IF MID$(U$,I,2)="I " THEN 5800
5790 A$=A$+MID$(U$,I,1):NEXT:RETURN
5800 A$=A$+"YOU":I=I+1
5810 GOTO 5780
5820 GOSUB 980:A=VAL(A$)
5830 REM REDO STRING CONTENTS
5840 R$=X$(45)
5850 GOTO 1220
```

Variables and Strings

N$ - First/last name of user
S$ - To differ between male and female
A$ - Used throughout the program for string entrys (input)
P$ - Entry of troubles
W - Word count, number of words in sentence (P$)
Y - Beginning of program, for use of periods in sentence
F$ - First name of user
TI - Time loop (pause) during program run
X$(N) - Data list of Words that computer can react to from the data statements.
I - Used as a counter
Q - Used in conjunction with A$(N) for the separation of words in a sentence
L - Number of letters in each word (LEN amount)
A(N) - Total number of letters per word
Z - Total count of words
W$ - Singled-out word from the beginning sentence that matches a word from the data list
M - Location of a particular word from the data list
A - VALue of A$
R$ - Singled-out word from the data list
KL - 0/1 variable for messages, comparison
DA$ - Length of troubles, i.e., days, weeks, months, or years
JK,KJ - 0/1 variable (true/false comparison)
Z - A random amount for print statements
JJ - Also for print statements
E$ - Contents of A$ (selected areas)
X - For random print statements
JX - 0/1 variable
U - Used in the for-next loop (A$) that examines the singled-out word
TT - Secondary time loop (pause) for end

U$ - Contents compared, for changing "I" to "YOU" for print messages

Explanation of the Program Lines

Lines 10-145 clear memory for string storage and display brief instructions.

Lines 150-160 request the input of the user's first and last name.

Lines 170-210 begin the actual program if A$ equals "Y"

Line 320 contains the dimension statements.

Lines 325-465 are used for the input of P$ (troubles sentence), the determination of the number of words in that sentence, and the display of messages if there are too many words (line 440) or too few (line 445).

Lines 490-540 separate N$ to set F$ equal to the first name of the user.

Lines 550-570 is the time loop (pause).

Lines 580-590 read the data list and place the words in the X$ array.

Lines 600-780 separate each word in the sentence entered by the user so that the individual words can be compared with the words in the data list.

Lines 790-970 compares each word in the sentence with those in the data list, X$(N). If no match is located, the program will branch to the print statements in lines 930-960.

Line 980 is a routine for the entry of A$ used by GOSUB statements throughout the program.

Line 990 enables the user to enter another trouble sentence.

Line 1000 terminates the program if A$ is equal to "NO."

Lines 1010-1120 direct the program flow to the appropriate line depending on which word in the data list matches a word in the sentence entered by the user.

Line 1190 causes a branch to a different set of print statements if any of the conditions are met.

Line 1200 is the beginning of all the print statements, questions, and possible solutions.

Lines 1205-4530 are all the print statements, inputs, and solutions, depending on the word(s) selected (as they are matched with computer's data list)

Lines 4540-4595 alter words such as *him, her, husband,* and *wife* according to the contents of W$

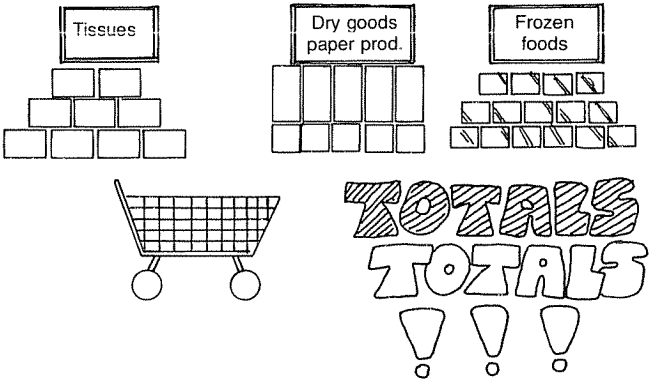
Lines 4600-4640 terminate the program if A$ is not equal to "Y" or "YES".

Lines 4700-4870 are extra print statements for use when the user becomes contrary or tries to confuse computer.

Lines 5200-5310 contain the data elements, which are the words that are compared to the individual words in the user's trouble sentence.

Lines 5740-5810 redo parts of sentence for the output of YOU as opposed to I.

Lines 5830-5850 changes the contents of R$ to match print output.

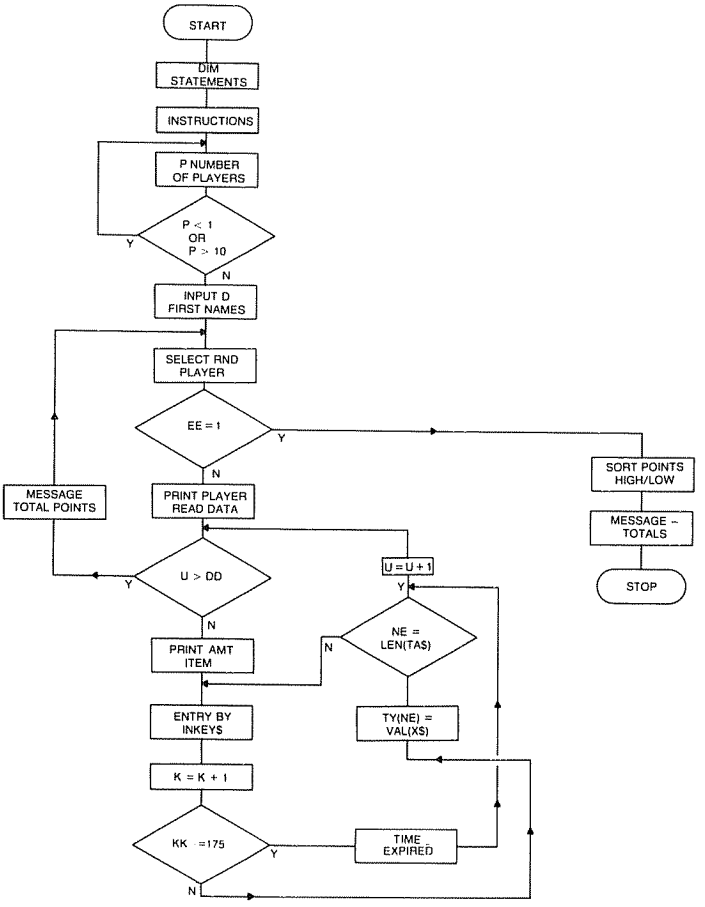


GENERAL STORE

So you think this is just a simple program to test your addition? Guess again! Sure, you'll have to be good at arithmetic and you'll have to add quickly! The computer will give you products (two at the first) and their prices. Then you'll have to enter the totals of the prices. All of this will happen in limited amount of time so, if you can't think fast...don't try this program! Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
GENERAL STORE
YOUR INSTRUCTIONS ARE SIMPLE AND
STRAIGHT-FORWARD: I WILL GIVE
YOU A LIST OF DIFFERENT ITEMS
AND THEIR PRICES. YOU IN TURN
WILL ENTER THE TOTAL OF THE
ITEMS. THAT IS, IF I SAY $2.50
MILK, THEN I SAY $0.80 BREAD,
YOU MUST THEN ENTER THE TOTAL
OF $2.50 + $0.80 ($3.30). ALL
YOUR ENTRIES WILL BE THROUGH
INKEY$, DON'T PRESS ENTER AND DON'T ENTER THE
DOLLAR SIGN.
PRESS A KEY...
ENTER ONLY THE NUMBERS AND THE
DECIMALS. AFTER 10 ITEMS WE WILL
```



Flowchart for *General Store*.

TAKE A SHORT BREAK: THEN I WILL GIVE YOU 10 MORE. WE WILL QUIT ONLY AFTER 30 ITEMS HAVE BEEN COMPLETED. YOUR SCORE WILL BE DETERMINED BY YOUR TOTAL. IF CORRECT, THE TOTAL IS YOUR SCORE. IF INCORRECT, YOUR TOTAL WILL BE A MINUS OF ALL ENTRIES MISSED. I SHOULD ALSO TELL YOU, THAT YOU WILL HAVE A TIME LIMIT IN WHICH TO ENTER THE TOTAL, SO LAY DOWN THE PENCIL AND CALCULATOR.

PRESS ANOTHER KEY...
HOW MANY PLAYERS? 2
ENTER 2 PLAYERS FIRST NAMES
MAX
MAY
I WILL SELECT EACH PLAYER IN
A RANDOM FASHION, THERE WILL
BE NO SET ORDER OF PLAYERS.

MAX YOU WILL NOW PLAY.
PRESS THE ENTER KEY...

Program Listing

```
 10 REM PROGRAM TITLE: GENERAL STORE
 20 CLS:CLEAR 500:PRINT TAB(10);
 30 PRINT"GENERAL STORE":R$="NEXT"
 40 RANDOM:DIM N$(15),I$(31),I(31)
 50 DIM L(31),PR(31),TP(15):REM INSTRUCTIONS
 60 PRINT"YOUR INSTRUCTIONS ARE SIMPLE AND"
 70 PRINT"STRAIGHT-FORWARD: I WILL GIVE"
 80 PRINT"YOU A LIST OF DIFFERENT ITEMS"
 90 PRINT"AND THEIR PRICES. YOU IN TURN"
100 PRINT"WILL ENTER THE TOTAL COST OF THE"
110 PRINT"ITEMS. THAT IS, IF I SAY $2.50"
120 PRINT"MILK, THEN I SAY $0.80 BREAD"
130 PRINT"YOU MUST THEN ENTER THE TOTAL"
140 PRINT"OF $2.50 + $0.80 ($3.30). ALL"
150 PRINT"YOUR ENTRIES WILL BE THROUGH"
160 PRINT"INKEY$. DON'T PRESS ENTER AND"
170 PRINT"DON'T ENTER THE DOLLAR SIGN"
180 PRINT:PRINT"PRESS A KEY..."
190 GOSUB 3000:CLS:G$="##,###"
200 PRINT"ENTER ONLY THE NUMBERS AND THE"
210 PRINT"DECIMALS. AFTER 10 ITEMS WE WILL"
220 PRINT"TAKE A SHORT BREAK. THEN I WILL"
230 PRINT"GIVE YOU 10 MORE. WE WILL QUIT"
240 PRINT"ONLY AFTER 30 ITEMS HAVE BEEN"
250 PRINT"COMPLETED. YOUR SCORE WILL BE"
260 PRINT"DETERMINED BY YOUR TOTAL, IF"
270 PRINT"CORRECT  THE TOTAL IS YOUR SCORE."
280 PRINT"IF INCORRECT, YOUR TOTAL WILL BE"
290 PRINT"A MINUS OF ALL ENTRIES MISSED."
300 PRINT"I SHOULD ALSO TELL YOU  THAT"
310 PRINT"YOU WILL HAVE A TIME LIMIT IN"
320 PRINT"WHICH TO ENTER THE TOTAL, SO LAY"
330 PRINT"DOWN THE PENCIL AND CALCULATOR."
```

```
340 REM BEGIN
350 PRINT"PRESS ANOTHER KEY..."
360 GOSUB 3000:CLS
370 INPUT"HOW MANY PLAYERS (1-10)";P
380 IF P<1 OR P>10 THEN 370
390 PRINT"ENTER";P;"PLAYER'S FIRST NAMES"
400 REM NAMES
410 AP=15538:FOR I=1 TO P
420 INPUT N$(I):N(I)=I:NEXT:N=1:AA=176
425 AQ=AP+64:IF P=1 THEN Y=1:N=2:GOTO 470
430 PRINT"I WILL SELECT EACH PLAYER IN"
440 PRINT"A RANDOM FASHION. THERE WILL"
450 PRINT"BE NO SET ORDER OF PLAYERS."
460 GOSUB 2800:PRINT:REM PLAYER
470 IF EE=1 THEN 1530:REM FINAL
480 Q$=N$(Y):TA=0:JI=0
490 PRINT Q$;" YOU WILL NOW PLAY."
500 PRINT"PRESS THE ENTER KEY...";
510 INPUT X$:PRINT
520 PRINT"HOW FAST IS YOUR ADDITION ";
530 PRINT Q$;" ???"
540 REM READ ALL ITEMS / PRICES
550 FOR I=1 TO 30:READ I$(I):L(I)=I:NEXT
560 FOR I=1 TO 30:READ PR(I):NEXT:GOTO 2500
570 D=I-1:DD=(D/3):U=1:D$="$##.##":F$="$"
580 REM SELECT ITEMS RANDOMLY
590 IF U>DD THEN PRINT:PRINT:GOTO 1100
600 X=RND(30)
610 IF L(X)=0 THEN 600:REM USED
620 T=X:PR=PR(T):T$=I$(T)
625 IF U=1 THEN YU=PR
630 TA=TA+PR:REM TOTAL AMOUNT/CURRENT ITEM
640 TA$=STR$(TA):REM TOTAL TO STRING
645 IF U=1 THEN GOSUB 2900
650 IF U=1 THEN CLS ELSE IF U>1 THEN 670
660 PRINT TAB(0)"PRICE" TAB(25)"ITEMS";
665 PRINT TAB(50)"TOTALS"
670 PRINT TAB(0) USING D$;PR;
680 PRINT TAB(25) T$
690 IF U<=1 THEN 710
700 RR=1:REM CONTINUE PRINT / ENTRIES
710 L(X)=0:U=U+1:KK=0:NE=1
720 IF RR<>1 THEN 580
725 PRINT@AA,F$
730 X$=INKEY$:IF KK<=5 AND X$<>"" THEN 1040
735 IF X$="." THEN 880
740 IF X$="" THEN 760
750 GOTO 800
```

```
 760 KK=KK+1:REM TIME EXPIRED ?
 770 IF KK>=175 THEN W$="":GOTO 1000
 780 POKE AP,143:GOTO 730
 800 REM ENTRY
 810 POKE AP,ASC(X$):TY(NE)=VAL(X$)
 820 AP=AP+1:REM POKE LOCATION
 830 NE=NE+1:REM NUMBER ENTRY
 840 IF NE=LEN(TA$) THEN 860
 850 GOTO 730
 860 GOTO 910:REM ALL ENTRIES IN
 880 REM DECIMAL
 890 POKE AP,ASC(X$)
 900 TY(NE)=1000:GOTO 820
 910 REM COMPILE ENTRIES
 920 V=NE-1:IF V=9 THEN V=10
 930 FOR I=1 TO V
 940 IF TY(I)=1000 THEN 970
 950 W$=W$+STR$(TY(I))
 960 TY(I)=0:NEXT:GOTO 990
 970 W$=W$+"."
 980 GOTO 960
 990 W=VAL(W$):W$=STR$(W)
1000 IF W$<>TA$ THEN PI=PI+PR:JI=JI+1
1005 IF U=3 AND JI<>0 THEN PI=PI+YU:JI=JI+1
1008 IF PEEK(AP)=143 THEN POKE AP,32
1010 POKE AP+1,32:W$="":AA=AA+64
1020 AP=AQ:AQ=AQ+64
1030 GOTO 590:REM ANOTHER ITEM
1040 REM SLOW DOWN A LITTLE
1050 POKE AP,32
1060 GOTO 730
1100 REM POINTS / PLAYER ADVANCE
1110 IF PI<>0 THEN 1200
1120 FOR I=1 TO LEN(TA$)
1130 IF MID$(TA$,I,1)="." THEN 1150
1140 K$=K$+MID$(TA$,I,1)
1150 NEXT
1160 K=VAL(K$)
1170 GOSUB 1430
1180 TP(Y)=TP(Y)+K
1190 GOTO 1320
1200 REM MISSED ITEM(S)
1210 TA$=STR$(PI):FOR I=1 TO LEN(TA$)
1220 IF MID$(TA$,I,1)="." THEN 1240
1230 K$=K$+MID$(TA$,I,1)
1240 NEXT:K=VAL(K$):GOSUB 1430
1250 TP(Y)=TP(Y)-K
1260 PRINT"YOU HAVE MISSED TOTALS ON";JI
1270 PRINT"ITEMS ";Q$;". THESE ITEMS CAME"
```

```
1280 PRINT"TO A TOTAL OF $";TA$;". YOU ARE"
1290 PRINT"MINUS THAT NUMBER, YOUR SCORE"
1300 PRINT"IS NOW: ";USING G$;TP(Y)
1310 GOTO 1350
1320 PRINT"YOU HAVE SUCCESSFULLY ENTERED"
1330 PRINT"ALL TOTALS ";Q$;", YOUR SCORE
1340 PRINT"IS NOW: ";USING G$;TP(Y)
1350 K=0:JI=0:PI=0:TA$="":K$=""
1360 U=1:D1=D1+10:IF D1=20 THEN R$="LAST"
1370 AP=15538:AQ=AP+64:AA=176:RR=0:PRINT
1380 IF D1>=30 THEN 1500
1390 PRINT"TAKE A SHORT BREAK, THEN WE'LL"
1400 PRINT"ATTEMPT THE ";R$;" SET OF ITEMS."
1410 FOR I=1 TO 5000:NEXT:PRINT
1420 GOTO 580
1430 REM AMOUNT TIMES 10
1440 IF MID$(TA$,LEN(TA$)-1,1)="." THEN 1480
1450 IF LEN(TA$)=2 OR LEN(TA$)=3 THEN 1470
1460 RETURN
1470 K=K*100:GOTO 1460
1480 K=K*10
1490 GOTO 1460
1500 REM ANOTHER PLAYER
1510 PRINT:PRINT:RESTORE
1520 D1=0:R$="NEXT":GOTO 460
1530 REM HIGH POINT
1535 IF P=1 THEN TP=TP(1):GOTO 1720
1540 MM=0:VV=1:N=N-1
1550 IF TP(VV)>=TP(VV+1) THEN 1590
1560 TP=TP(VV):TP(VV)=TP(VV+1)
1570 TP(VV+1)=TP:N$=N$(VV)
1580 N$(VV)=N$(VV+1):N$(VV+1)=N$:MM=1
1590 IF VV+1>=N THEN 1600:REM FINAL
1595 VV=VV+1:GOTO 1550
1600 REM MM=1 -- NOT IN ORDER
1610 IF MM=1 THEN 1540
1650 REM POINTS COMPILED (FINAL)
1700 REM PRINT RESULTS
1710 PRINT"ALL HAVE PLAYED...."
1720 PRINT"PRESS A KEY FOR RESULTS ON SCORE"
1730 GOSUB 3000
1740 CLS:I=1
1750 REM SINGLE PLAYER
1760 IF P=1 THEN 1850
1770 PRINT"THE TOP WINNER: ";N$(I)
1780 PRINT"WITH A FINAL POINT SCORE OF: ";
1790 PRINT USING G$;TP(I)
1800 PRINT"OTHER PLAYERS AND SCORES:"
1810 FOR X=I+1 TO P
```

```
1820 PRINT N$(X);" HAD: ";USING G$;TP(X)
1830 NEXT
1840 GOTO 1880
1850 PRINT Q$;" YOUR TOTAL (FINAL) SCORE"
1860 PRINT"FOR GENERAL STORE IS: ";
1870 PRINT USING G$;TP
1880 PRINT
1890 PRINT"END OF PROGRAM RUN."
1900 END
2500 REM INFLATION / DEFLATION
2510 FOR I=1 TO 30
2520 DE=RND(10):DF=CSNG(DE/10)
2530 DG=RND(2)
2540 IF DG=2 THEN 2590
2550 REM INFLATION
2560 PR(I)=PR(I)+DF
2570 NEXT
2580 GOTO 570
2590 REM DEFLATION
2600 PR(I)=PR(I)-DF
2610 IF PR(I)<=0 THEN 2560
2620 GOTO 2570
2800 REM RANDOM PLAYER
2810 IF N>P THEN EE=1:RETURN
2820 Q=RND(P)
2830 IF N(Q)=0 THEN 2820
2840 Y=Q:N(Q)=0
2850 N=N+1:RETURN
2900 REM EXTRA MESSAGE
2910 PRINT
2920 PRINT"ADD ALL ITEMS TOGETHER"
2930 PRINT"BEFORE INPUTTING AMOUNT."
2940 IF D1>=10 THEN 2970
2950 FOR TL=1 TO 2800:NEXT
2960 CLS:RETURN
2970 PRINT"START WITH PREVIOUS TOTAL."
2980 GOTO 2950
3000 REM INKEY$
3010 X$=INKEY$
3020 IF X$="" THEN 3010
3030 RETURN
4000 REM DATA / PRICE - ITEM COMBINED
4010 DATA MILK,BREAD,MEAT
4020 DATA MEAT,MEAT,FISH
4030 DATA CANDY,CANDY,CANDY
4040 DATA VEGETABLES,VEGETABLES
4050 DATA POULTRY,POULTRY
4060 DATA LETTUCE,ONIONS
```

```
4070 DATA SPINACH,CIGARETTES
4080 DATA CIGARS,BACON
4090 DATA TABLETS (PAPER),PENS
4100 DATA BATTERIES,BATTERIES
4110 DATA FURNITURE POLISH
4120 DATA CARPET CLEANER
4130 DATA HAND SOAP,HAND SOAP
4140 DATA CRACKERS,MEATS
4150 DATA SODAS (CASE)
4160 REM PRICES OF ITEMS
4170 DATA 2.30,0.89,1.98,3.35,5.50,3.65
4180 DATA 0.35,2.05,0.54,0.39,0.45,3.44
4190 DATA 1.99,0.59,1.01,0.79,7.05,5.74
4200 DATA 4.66,1.23,2.44,2.66,0.98,1.97
4210 DATA 2.41,0.99,0.99,1.32,8.99,4.32
```

Variables and Strings

G$ - Used for the printing of scores
P - Number of players (1-10)
AP - Poke location for printing
N$(N) - First names of players
N - Number of players, counter
AA - Print @ locations
AQ - Same as AP, but for advancing to a new area
Q$ - Selected player
TA - Total amount, up to and including current item
JI - Counter for incorrect price entry
I$(N) - All items
PR(N) - Price of above items
D - Amount of I after data is read
DD - Total items
U - Counter for items
D$ - For print of prices
X - Random selection of an item
L(X) - Delete the item X, after selection
T - Same as variable X (random)
PR - Set to the price of the current item, PR(T)
T$ - Same as the I$(T) item, used in conjunction with JI
YU - Price of the first item in each set
TA$ - STR$ of amount TA
KK - Used for timing responses
NE - Used for TY (subscripted) for each numeral entry
TY(N) - Explained in the line above
W$ - STR$ content of the TY(I) cost entry

PI - For incorrect price entry
K$ - Total cost of 10 items
K - VALue of K$, above
TP(Y) - Total score for current player
VV - Used for sorting high to low score list (if more than one has played)
MM - 0/1, true/false variable. If MM= 1, the list of players arranged according to their scores is not correct in order. Likewise if MM = 0 the list is correct
DE - Selects a random amount from 1-10 for inflation or deflation of current prices
DF - Insures that the amount above will be a "cent" increase of decrease
DG - Determines whether the current price in the list will be inflated or deflated
Q - Randomly selected player

Explanation of the Program Lines

Lines 40-50 dimension the strings and variables.

Lines 60-350 display the instructions for play.

Line 360 contains a GOSUB to line 3000, which contains an INKEY$ routine that line 360 uses to exit the instructions. This routine is used throughout the program

Lines 370-420 let the user enter the number of players and their first names.

Lines 410, 420, and 425 also initialize some variables for print@ (AP) and poke (AP and AQ) locations.

Lines 430-460 print a message, and then select one player randomly using a GOSUB to lines 2800-2850. If only one person is playing, P=1, and line 425 causes the random selection of a player to be skipped.

Line 470 if EE=1, all players have had their turn. The program will branch to the final messages beginning at line 1530.

Line 480 sets the contents of Q$ to the name of the player selected, N$(Y). The variables TA and JI are set or reset to 0 for totaling and scoring purposes.

Lines 490-530 are print messages for the current player.

Lines 540-560 read all data items and prices. The subscripted variable L(I) is used to delete a particular item after it has been used. After all items I$(I) and prices PR(I) are read, the program will branch to lines 2500-2620. These lines are used for inflation or deflation of the 30 prices. This will insure that the user cannot

memorize any price because each will change with each player or each program run. Line 2520 randomly selects a number from 1-10 (DE). This number is then divided by 10 and set to a cent amount, DF = C(SNG(DE/10). The CSNG simply insures, that DF is a single-precision number. Line 2530, where DG = RND(2), will determine whether this amount is to be used to increase or decrease the current price. If DG is equal to 2, the current amount will be decreased by DF (lines 2590-2620). If the amount of PR(I) is less than or equal to 0 (zero), the program will branch back and use DF to increase PR(I) in line 2560. The current price will always be a positive amount, even if it's only one penny.

Line 570 sets more variables and strings: DD for the total number of items (D) divided by 3; (the program presents 3 sets of 10 items) U, a counter for the items; and D$ and F$ for print statements of prices.

Lines 580-610 select one item at random if U is not greater than DD (line 590) and the item has not been omitted (line 610).

Lines 620-640 select an item and the total price of the item selected.

Lines 645-650 are used primarily for print statements if U is equal to one (line 660 and 665).

Line 670-680 print the price and the item selected.

Lines 690-700 insure that two items are selected if U<=1, so that the user will have a total to enter.

Line 710 deletes the selected item, increases the item counter, U, sets the timer counter, KK, and sets variable NE to 1 for the contents of key entries.

Line 720 branches back to line 580 if RR is not equal to 1, as explained in lines 690 and 700 above.

Line 725 prints the dollar sign at AA, so the user will know the timing has begun.

Lines 730-770 compose the INKEY$ routine for user entries. The variable KK is increased by 1 if no key closure is detected. The argument at line 730, if KK<=5 and X$<>"", will insure that the beginning of a new entry is cleared (lines 1040-1060) in case a key is closed when KK>=175. This way, the amount will not be carried over to the next item. If the decimal key is pressed (line 735), the program will branch to line 880; that ASCII value will be poked onto the video, and that part of the price entered will be equal to 1000, TY(NE). If a numeral key is pressed, the program will branch to line 800. That numeral will be poked at AP and TY(NE) will be equal to the value amount of X$. With this function and the above the poke

location is increased by 1 and the variable NE is also increased by 1. Line 840 tests the value of NE. If it is equal to LEN(TA$), the required number of keys have been pressed, and the program branches to line 910.

Line 780 pokes CHR$(143), a graphic block, at the current location of the price entry. If no numeral or decimal is pressed, lines 910-980 will compile all the keys closed, and place them in a string (W$) along with the decimal (line 940), if TY(I)=1000.

Line 990 sets W to the VALue of W$. W$ is then changed to the STR$ amount of W.

Line 1000 increases the variable PI to PR (the price of the item) if W$ is not equal to TA$. JI will then be increased by 1.

Line 1005 is used to include the price of the first item in the current set (YU), if the total entered to that point is incorrect and U= 3 and JI< > 0.

Line 1008 clears the cursor block (143) from the current line before advancing to the next item.

Line 1010 clears the contents of W$ (for another item), advances the print@area by 64 (AP=AQ), and increases AQ by 64 for the next poke location.

Lines 1020-1030 start the process all over by going back to line 590 to select another item. If the condition is met at line 590 (10 items have been completed), the program will branch to line 1100.

Line 1110 checks the amount of PI. If it is not equal to 0 (zero), the program will branch to the incorrect entries routine.

Lines 1120-1190 are used for correct entries. TA$ is torn apart and placed in another string, K$. If the decimal is encountered (line 1130) the program will branch to the next statement in line 1150. In line 1160 K will assssume the amount of K$, through the VAL function.

Line 1170 contains a GOSUB to line 1400, which will again check the contents of TA$ (lines 1440 and 1450). If either condition is met, the variable K will be multiplied by the appropriate amount, for the addition of trailing zeros.

Line 1180 increases the total points, TP(Y), by the value for each player.

Line 1190 then branches to line 1320 as explained below.

Lines 1200-1250 are used for missed items. They function in the same manner as lines 1120-1190, but the value of each item missed is subtracted from the player's total. The amount returned will be a negative number.

Lines 1260-1300 are the print messages displayed when items

have been missed. The number of incorrect items (JI), the total price, and the player's current score are displayed.

Lines 1320-1340 are the print messages displayed when no items have been missed. The player's score is displayed. Note that all items must have been entered correctly or the player will receive a negative score, even if only one was missed.

Lines 1350-1420 begin the cycle again for the next 10 items, if the condition is not met at line 1380. Variables are cleared and strings are set to null.

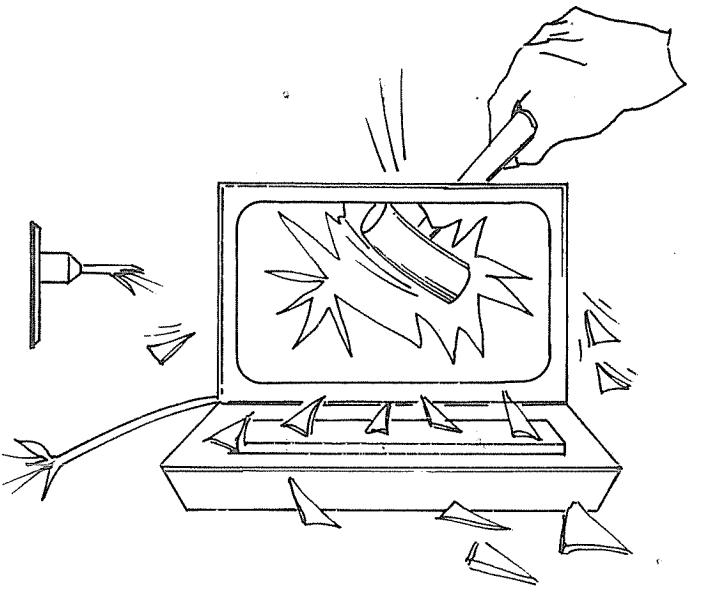
Lines 1430-1490 are explained above.

Lines 1500-1520 restore the data list for another reading before the selection of the next player, if any remain.

Lines 1530-1610 sort the scores, along with the appropriate players' names, from high to low if there is more than one player (line 1535). If MM is equal to 1 in line 1600 and 1610, the list is not perfect and the sort will begin again. Otherwise the program will continue to line 1650.

Lines 1650-1900 print all the players' names and their accumulated scores and terminate the program.

Lines 4000-4210 contain the data items and their corresponding prices.



RESPECT YOUR COMPUTER

Do you have respect for your computer? Do you believe that it is telling you the truth all of the time? You're in for a surprise then! This is an excellent program to try on that person or friend who thinks he or she is always right or who feels they never make a wrong guess. Keep all heavy objects away from your computer when running this program; your friends might be tempted to bash the CPU in!

Sample Run

```
NATURALLY YOU SHOULD HAVE SOME
RESPECT FOR YOUR COMPUTER, EVEN
THOUGH IT IS ONLY A MACHINE.
THIS MACHINE WILL NOW TEST YOUR
PATIENCE, TO SEE JUST HOW MUCH
RESPECT YOU DO HAVE FOR THIS
MACHINE. YOU SEE, THIS
IS ONE GAME THAT YOUR COMPUTER
WILL STAKE ITS CIRCUITS ON!
YOU CANNOT, AND WILL NOT
BEAT IT !!!
```

PRESS ENTER...PLEASE?

IF YOU ARE READY, WE WILL NOW
BEGIN, I HAVE SELECTED A NUMBER
BETWEEN 1 AND 10, TELL ME
THE NUMBER? 4
YOUR NUMBER: 4
SORRY.

NUMBERS GUESSED: 4

GUESS AGAIN...

THE NUMBER? 1
YOUR NUMBER: 1
SORRY.

NUMBERS GUESSED: 4 1

GUESS AGAIN...

THE NUMBER? 9
YOUR NUMBER: 9
SORRY.

NUMBERS GUESSED : 4 1 9
GUESS AGAIN...

THE NUMBER? 3
YOUR NUMBER: 3
SORRY.

NUMBERS GUESSED: 4 1 9 3

GUESS AGAIN...

THE NUMBER? 10
YOUR NUMBER: 10
SORRY.

NUMBERS GUESSED : 4 1 9 3 10

GUESS AGAIN...

THE NUMBER? 8
YOUR NUMBER: 8
SORRY.

NUMBERS GUESSED: 4 1 9 3 10 8

GUESS AGAIN...

THE NUMBER? 7
YOUR NUMBER: 7
SORRY.

NUMBERS GUESSED: 4 1 9 3 10 8 7

GUESS AGAIN...

THE NUMBER? 6
YOUR NUMBER: 6
SORRY.

NUMBERS GUESSED: 4 1 9 3 10 8 7 6

GUESS AGAIN...

THE NUMBER? 2
YOUR NUMBER: 2
SORRY.

NUMBERS GUESSED: 4 1 9 3 10 8 7 6 2

GUESS AGAIN...

NEVERMIND...
THE NUMBER I HAD SELECTED WAS: 5

YOUR GUESSES WERE: 4 1 9 3 10 8 7 6 2

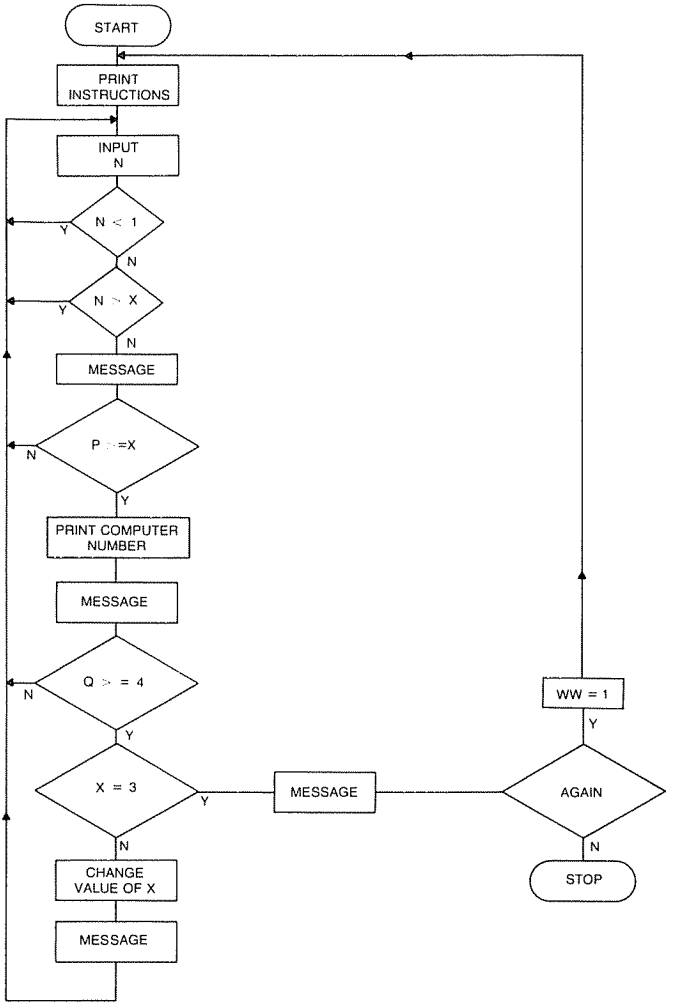
LET'S TRY AGAIN...

OF THE 9 GUESSES YOU MADE,

NONE OF THEM MATCHED THE
NUMBER I HAD SELECTED.....

STOP

This is not a complete run.



Flowchart for *Respect Your Computer*.

Program Listing

```
 100 REM PROGRAM TITLE: RESPECT YOUR COMPUTER
 150 CLS
 200 RANDOM:P=1:Q=1:X=10:IF WW=1 THEN 900
 250 PRINT"NATURALLY YOU SHOULD HAVE SOME"
 300 PRINT"RESPECT FOR YOUR COMPUTER, EVEN"
 350 PRINT"THOUGH IT IS ONLY A MACHINE."
 400 PRINT"THIS MACHINE WILL NOW TEST YOUR"
 450 PRINT"PATIENCE TO SEE JUST HOW MUCH"
 500 PRINT"RESPECT YOU DO HAVE FOR THIS"
 550 PRINT"MACHINE.  YOU SEE, THIS"
 600 PRINT"IS ONE GAME THAT YOUR COMPUTER"
 650 PRINT"WILL STAKE ITS CIRCUITS ON"
 700 PRINT"YOU CANNOT, AND WILL NOT"
 750 PRINT"BEAT IT !!!"
 800 PRINT
 850 INPUT"PRESS ENTER...PLEASE";XX
 900 CLS
 950 PRINT"IF YOU ARE READY, WE WILL NOW"
1000 PRINT"BEGIN.  I HAVE SELECTED A NUMBER"
1050 PRINT"BETWEEN 1 AND";X;".  TELL ME"
1100 INPUT"THE NUMBER";N
1150 IF N<1 OR N>X THEN 1100
1200 PRINT"YOUR NUMBER: ";N
1250 FOR I=1 TO 800:NEXT
1300 PRINT CHR$(83);CHR$(79);CHR$(82);
1350 PRINT CHR$(82);CHR$(89);CHR$(46)
1400 PRINT
1450 A(P)=N
1500 PRINT"NUMBERS GUESSED:";
1550 FOR I=1 TO P:PRINT A(I);:NEXT
1600 PRINT:P=P+1
1650 PRINT"GUESS AGAIN..."
1700 IF P>=X THEN 1900
1800 PRINT
1850 GOTO 1100
1900 PRINT
1950 PRINT"NEVERMIND..."
2000 PRINT"THE NUMBER I HAD SELECTED WAS:";
2050 A=X
2100 FOR I=1 TO P-1
2150 IF A<>A(I) THEN NEXT:GOTO 2250
2200 A=A-1:GOTO 2100
2250 REM PRINT NUMBER
2300 PRINT A:PRINT
2350 PRINT"YOUR GUESSES WERE:";
2400 FOR I=1 TO P-1
```

```
2450 PRINT A(I);:NEXT
2500 PRINT
2550 Q=Q+1:TP=TP+P-1
2600 IF Q>=4 THEN 3000
2650 PRINT"LET'S TRY ";
2700 IF Q=3 PRINT"ONCE MORE...":GOTO 2800
2750 PRINT"AGAIN..."
2800 FOR I=1 TO 1200:NEXT
2850 PRINT:P=1
2900 GOTO 850
3000 PRINT
3050 PRINT"OF THE";TP;"GUESSES YOU MADE,"
3100 PRINT"NONE OF THEM MATCHED THE"
3150 PRINT"NUMBER I HAD SELECTED....."
3200 FOR I=1 TO 2000:NEXT:Z=X
3250 REM MORE ?
3300 IF X=10 THEN X=5:GOTO 3450
3350 IF X=5 THEN X=3:GOTO 3450
3400 A=0:GOTO 3850
3450 PRINT:P=1:Q=1:IF X=3 GOTO 3600
3500 PRINT"TO TAKE THIS A STEP FURTHER,"
3550 PRINT"LET'S TRY SOMETHING DIFFERENT."
3600 PRINT"INSTEAD OF 1 AND";Z;"I WILL"
3650 PRINT"SELECT BETWEEN 1 AND";X;"AND"
3700 PRINT"GIVE YOU";X-1;"TRIES..."
3750 PRINT
3800 GOTO 850
3850 PRINT
3900 PRINT"THIS IS REALLY HARD TO BELIEVE..."
3950 PRINT"YOU MADE";TP;"GUESSES, I GAVE"
4000 PRINT"YOU THREE DIFFERENT BREAKS AND"
4050 PRINT"YOU MATCHED";A;"OF THE NUMBERS"
4100 PRINT"THAT I SELECTED !!!"
4150 FOR I=1 TO 3000:NEXT
4200 CLS:PRINT CHR$(23)
4250 PRINT"YOU CAN'T WIN....."
4300 PRINT:PRINT"TURKEY !!"
4350 FOR I=1 TO 1500:NEXT:CLS
4400 PRINT"READY TO TRY AGAIN, YOU'LL"
4450 PRINT"GET THE SAME BREAKS AS BEFORE";
4500 INPUT X$
4550 IF X$="N" OR X$="NO" THEN 4650
4600 WW=1:GOTO 100
4650 CLS:PRINT CHR$(23)
4700 PRINT"CHICKEN..."
4750 GOTO 4750
```

Variables and Strings

P - Attempts (guesses)
X - Attempts per run
N - Number input as guess
A(I) - Subscripted variable for guesses
A - Computer selected number
TP - Total guesses
Z - Value of X (current) for message
WW - For skipping the instructions (playing more than once)

Explanation of the Program Lines

Line 200 initializes three variables and skips the instructions if WW is equal to 1 (more than one run).

Lines 250-850 print the general instructions.

Lines 900-1100 begin the rounds and accept user's guess.

Line 1150 tests the value of N to make sure it is within the specified limits.

Lines 1200-1550 print several messages and the number guessed. It then places each number in a subscript A(P) = N for further reference.

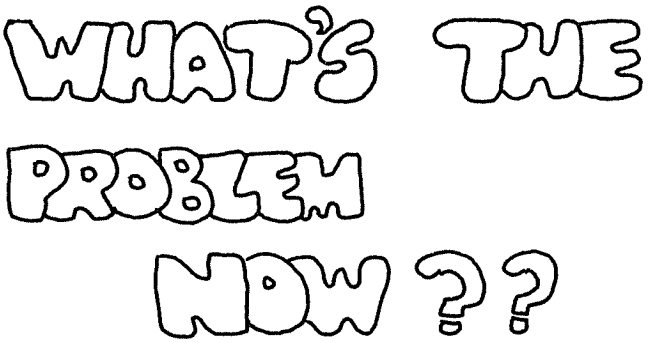
Lines 1600-1850 cause a return to line 1100 for another guess if P is not greater than or equal to X.

Lines 1900-2450 print the number selected by the computer and all of the numbers the user selected.

Lines 2500-2900 begin it all again if Q is less than or equal to 3 and the user has not hit the computer with a hammer.

Lines 3000-3800 print the user's total number of guesses, change the value of X to a lower number, and then start the routine again. If X = 3, the program branches to line 3850.

Lines 3850-4750 print a closing message and ask the user if he or she would like to try again.



LOVE ONE ANOTHER

Having problems with your spouse, girlfriend, or boyfriend? Don't take it to court, take it to your computer! If you're having difficulties expressing your complaint, you can register it with this program. Then your wife, husband, girlfriend, or boyfriend can register his or her complaints. The final outcome? Discussions of problems you thought didn't exist!

Sample Run

```
LOVE ONE ANOTHER

ARE YOU MARRIED? YES
WHAT IS YOUR SPOUSE'S
FIRST NAME, PLEASE? MARGRET
SO YOU ARE HAVING A FEW PROBLEMS
AND NEED YOUR COMPUTER'S ASSISTANCE?
YOU WILL GO THROUGH THE PROGRAM
FIRST; THEN MARGRET CAN GO THROUGH
IT. I WILL THEN GO THROUGH EACH
LIST AND PICK OUT THE MAJOR
ISSUES. IF AT ALL POSSIBLE, KEEP
COMPLAINTS TO THREE OR FEWER
WORDS. THIS WILL KEEP THINGS
EASIER TO MANAGE.
```

BEGIN?

YOUR FIRST NAME? SAM
SAM YOU WILL HAVE THE
OPPORTUNITY TO REGISTER
UP TO 10 COMPLAINTS. IF YOU
DO NOT HAVE 10, JUST PRESS
'ENTER' FOR A COMPLAINT.

LET'S HAVE COMPLAINT NUMBER 1 ?RELATIVES
NUMBER 2 ?COMMUNICATION
NUMBER 3 ?SMOKING
NUMBER 4 ?GOING PLACES
NUMBER 5 ?RELATIVES
NUMBER 6 ?

WE WILL NOW GIVE EACH
COMPLAINT A PRIORITY NUMBER.
THESE NUMBERS WILL RANGE
FROM 1 (LEAST SIGNIFICANT)
TO 10 (MOST SIGNIFICANT). NO
FRACTIONS, PLEASE.

COMPLAINT NUMBER 1 :
RELATIVES...PRIORITY? 8
COMPLAINT NUMBER 2
COMMUNICATION...PRIORITY? 10
COMPLAINT NUMBER 3 :
SMOKING...PRIORITY? 6
COMPLAINT NUMBER 4 :
GOING PLACES...PRIORITY? 7
COMPLAINT NUMBER 5:
RELATIVES...PRIORITY? 8

PROCESSING DATA...

ALL COMPLAINTS HAVE BEEN
PROCESSED AND PLACED IN
PRIORITY ORDER, SAM.

IF YOU WILL CALL MARGRET TO
THE COMPUTER, I WILL ACCEPT

THE 'OTHER SIDE' OF THE
COIN.....
MARGRET PRESS ENTER?
MARGRET, SAM HAS ENTERED 5
COMPLAINTS. I WILL NOW RECEIVE
YOUR VERSION OF COMPLAINTS, AND
THEN COMPARE THE TWO SETS
AND RETURN MY FINDINGS...

MARGRET YOU WILL HAVE THE
OPPORTUNITY TO REGISTER UP
TO 10 COMPLAINTS, IF YOU
DO NOT HAVE 10, JUST PRESS
'ENTER' FOR A COMPLAINT.

LET'S HAVE COMPLAINT NUMBER 1 ?RELATIONS

NUMBER 2?

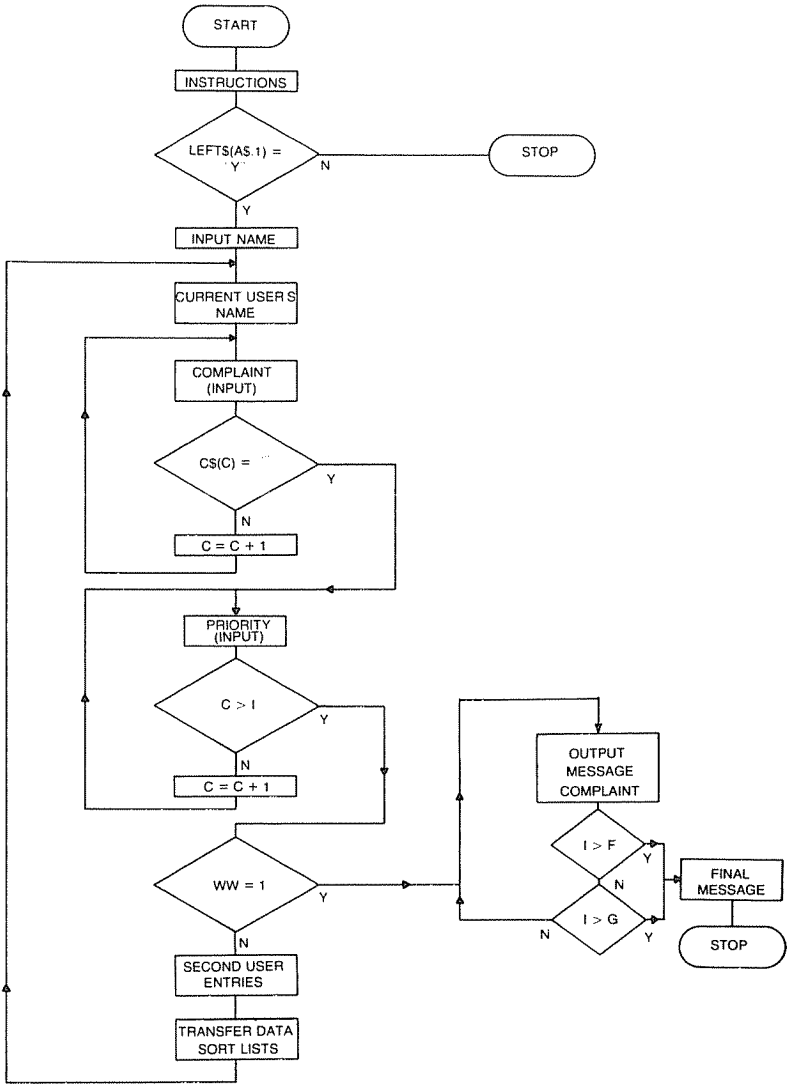
WE WILL NOW GIVE EACH
COMPLAINT A PRIORITY NUMBER.
THESE NUMBERS WILL RANGE
FROM 1 (LEAST SIGNIFICANT)
TO 10 (MOST SIGNIFICANT). NO
FRACTIONS, PLEASE.

COMPLAINT NUMBER 1:
RELATIONS...

PRIORITY? 10

PROCESSING DATA...

ALL COMPLAINTS HAVE BEEN
PROCESSED AND PLACED IN
PRIORITY ORDER MARGRET.
SAM HAS MORE
COMPLAINTS, AS YOU PROBABLY
SEE IN THE PRINTOUT, MARGRET.
BUT THAT ISN'T THE ISSUE.
WHAT IS IMPORTANT ARE THE
MAJOR COMPLAINTS:
SAM'S IS: COMMUNICATION
MARGRET'S IS: RELATIONS



Flowchart for *Love One Another*.

```
DISCUSS THIS PROBLEM BETWEEN
YOURSELVES NOW, THEN PRESS ENTER?

MARGRET HAS NO MORE COMPLAINTS.
SAM'S OTHER COMPLAINTS ARE:
RELATIVES
DISCUSS THIS PROBLEM BETWEEN
YOUSELVES NOW, THEN PRESS ENTER?

MARGRET HAS NO MORE COMPLAINTS.
SAM'S OTHER COMPLAINTS ARE:
GOING PLACES


STOP
```

This is not a complete run. In between the discussions of the complaints, each is placed on a chart for viewing.

Program Listing

```
 010 REM PROGRAM TITLE: LOVE ONE ANOTHER
 20 CLEAR 1000:CLS
 30 PRINT"LOVE ONE ANOTHER":PRINT
 40 INPUT"ARE YOU MARRIED";A$
 50 IF LEFT$(A$,1)="Y" THEN 120
 60 PRINT"ARE YOU CURRENTLY DATING A"
 70 INPUT"GIRLFRIEND OR BOYFRIEND";A$
 80 IF LEFT$(A$,1)="Y" THEN 160
 90 PRINT"WHY ARE YOU RUNNING THIS"
100 PRINT"PROGRAM  DINGY !!!"
110 GOTO 1840
120 PRINT"WHAT IS YOUR SPOUSE'S"
130 PRINT"FIRST NAME, PLEASE";
140 INPUT N$
150 PRINT@0,CHR$(30);:GOTO 180
160 PRINT"WHAT IS YOUR GIRLFRIEND OR"
170 PRINT"BOYFRIEND'S FIRST NAME";:GOTO 140
180 PRINT"SO YOU ARE HAVING A FEW PROBLEMS"
190 PRINT"AND NEED YOUR COMPUTER'S ASSISTANCE."
200 PRINT"YOU WILL GO THROUGH THE PROGRAM"
210 PRINT"FIRST, THEN ";N$;" CAN GO THROUGH"
220 PRINT"IT.  I WILL THEN GO THROUGH EACH"
230 PRINT"LIST AND PICK OUT THE MAJOR"
240 PRINT"ISSUES.  IF AT ALL POSSIBLE, KEEP"
250 PRINT"COMPLAINTS TO LESS THAN FOUR"
```

```
260 PRINT"WORDS  THIS WILL KEEP THINGS"
270 PRINT"EASIER TO MANAGE."
280 PRINT
290 PRINT
300 INPUT"BEGIN";A$
310 CLS:DIM C$(11),P(11)
320 INPUT"YOUR FIRST NAME";NN$
330 PRINT NN$;" YOU WILL HAVE THE OPPORTUNITY"
340 PRINT"TO REGISTER UP TO TEN COMPLAINTS,IF YOU"
350 PRINT"DO NOT HAVE 10, JUST PRESS"
360 PRINT"'ENTER' FOR A COMPLAINT."
370 PRINT
380 C=1
390 PRINT"LET'S HAVE COMPLAINT NUMBER";C;
400 IF C>=2 PRINT"NUMBER ";C;
410 INPUT C$(C):GOTO 1220
420 IF C$(C)="" THEN 470
430 REM STEP COUNTER
440 C=C+1
450 IF C=11 THEN 470
460 GOTO 400
470 I=C-1
480 IF I=0 THEN 370
490 CLS
500 PRINT"WE WILL NOW GIVE EACH"
510 PRINT"COMPLAINT A PRIORITY NUMBER."
520 PRINT"THESE NUMBERS WILL RANGE"
530 PRINT"FROM 1 (LEAST SIGNIFICANT)"
540 PRINT"TO 10 (MOST SIGNIFICANT), NO"
550 PRINT"FRACTIONS, PLEASE."
560 PRINT
570 C=1
580 PRINT"COMPLAINT NUMBER";C;":"
590 PRINT C$(C);"...";
600 INPUT"PRIORITY";P(C)
610 IF P(C)<1 OR P(C)>10 THEN 630 ELSE C=C+1
620 IF C>I THEN 650
630 CLS
640 GOTO 580
650 REM ARRANGE AS PER PRIORITY
655 PRINT"PROCESSING DATA..."
660 C=1:X=0
670 IF P(C)>=P(C+1) THEN 710
680 P=P(C):P(C)=P(C+1):P(C+1)=P
690 C$=C$(C):C$(C)=C$(C+1):C$(C+1)=C$
700 X=1
710 C=C+1
720 IF C>I THEN 740
```

```
 730 GOTO 670
 740 IF X=1 THEN 660:REM ORDER AGAIN
 750 REM IN ORDER
 760 PRINT"ALL COMPLAINTS HAVE BEEN"
 770 PRINT"PROCESSED AND PLACED IN"
 780 PRINT"PRIORITY ORDER,  ";NN$;"."
 790 IF WW=1 THEN 1010
 800 WW=1
 810 PRINT
 820 PRINT"IF YOU WILL CALL ";N$;" TO"
 830 PRINT"THE COMPUTER, I WILL ACCEPT"
 840 PRINT"THE 'OTHER SIDE' OF THE"
 850 PRINT"COIN....."
 860 PRINT N$;:INPUT" PRESS ENTER";X
 870 CLS
 880 PRINT N$;", ";NN$;" HAS ENTERED";I
 890 PRINT"COMPLAINTS, I WILL NOW RECEIVE"
 900 PRINT"YOUR VERSION OF COMPLAINTS,"
 910 PRINT"THEN COMPARE THE TWO SETS"
 920 PRINT"AND RETURN MY FINDINGS..."
 930 REM TRANSFER DATA
 940 FOR W=1 TO I
 950 W$(W)=C$(W):C$(W)=""
 960 W(W)=P(W):P(W)=0
 970 NEXT
 980 X$=NN$:NN$=N$:F=I
 990 PRINT
1000 GOTO 330
1010 REM ALL COMPLAINTS IN
1020 PRINT
1030 INPUT"ENTER PLEASE";X
1040 CLS
1050 PRINT TAB(10) X$; TAB(45) N$
1060 PRINT TAB(0)"COMPLAINT";
1070 PRINT TAB(18)"PRIORITY";
1080 PRINT TAB(35)"COMPLAINT";
1090 PRINT TAB(52)"PRIORITY"
1100 FOR Q=0 TO 59:PRINT"X";:NEXT:PRINT
1110 G=I:I=1
1120 PRINT TAB(0) W$(I); TAB(21) W(I);
1130 IF I>G THEN 1150
1140 PRINT TAB(35) C$(I); TAB(56) P(I)
1150 I=I+1
1160 IF I<=F THEN 1200
1170 IF I>F AND I<=G THEN 1130
1180 REM BRANCH TO RESULTS
1190 GOTO 1300
1200 IF I>G THEN PRINT:GOTO 1120
```

```
1210 GOTO 1120
1220 REM LETTER COUNT (LENGTH)
1230 IF LEN(C$(C))>=21 THEN 1250
1240 GOTO 420
1250 PRINT"YOU NEED TO SHORTEN"
1260 PRINT"THE COMPLAINT A LITTLE."
1270 GOTO 400
1300 REM RESULTS
1310 PRINT:INPUT"KEY ENTER WHEN READY";X
1320 CLS
1330 IF F>G THEN PRINT X$;:R=1:GOTO 1360
1340 IF F=G THEN 1410
1350 PRINT N$;
1360 PRINT" HAS MORE"
1370 PRINT"COMPLAINTS, AS YOU PROBABLY"
1380 PRINT"SEE IN THE PRINTOUT ";
1390 IF R=1 PRINT N$;".":GOTO 1440
1400 PRINT X$;".":GOTO 1440
1410 PRINT X$;" YOU AND ";N$;" HAD AN"
1420 PRINT"EQUAL AMOUNT OF COMPLAINTS, AS"
1430 PRINT"YOU PROBABLY KNOW BY NOW."
1440 PRINT:I=1
1450 PRINT"BUT THAT ISN'T THE ISSUE."
1460 PRINT"WHAT IS IMPORTANT ARE THE"
1470 PRINT"MAJOR COMPLAINTS:"
1480 PRINT X$;"'S IS: ";W$(I)
1490 PRINT N$;"'S IS: ";C$(I)
1500 PRINT
1510 PRINT"DISCUSS THIS PROBLEM BETWEEN"
1520 PRINT"YOURSELVES NOW, THEN PRESS ENTER";
1530 INPUT X:CLS
1540 I=I+1
1550 IF I>F THEN 1600
1560 IF I>G THEN 1630
1570 PRINT:IF JJ=1 THEN 1650
1580 PRINT"THE NEXT AREA OF COMPLAINTS:"
1590 GOTO 1480
1600 PRINT X$;" HAS NO MORE COMPLAINTS."
1610 IF I>G THEN 1630
1620 GOTO 1650
1630 PRINT N$;" HAS NO MORE COMPLAINTS."
1640 GOTO 1690
1650 PRINT N$;"'S OTHER COMPLAINTS ARE:"
1660 JJ=1
1670 PRINT C$(I)
1680 GOTO 1500
1690 IF I>F THEN 1750
1700 PRINT X$;"'S OTHER COMPLAINTS ARE:"
```

```
1710 JJ=2
1720 PRINT W$(I)
1730 GOTO 1500
1750 REM FINISH
1760 PRINT
1770 PRINT"IF YOU DID NOT SOLVE ALL"
1780 PRINT"OF YOUR DISAGREEMENTS RUN"
1790 PRINT"THE PROGRAM AGAIN."
1800 PRINT
1810 PRINT"REMEMBER, WHEN YOU HAVE PROBLEMS"
1820 PRINT"DON'T TAKE IT TO COURT..."
1830 PRINT"TAKE IT TO YOUR COMPUTER."
1840 PRINT
1850 PRINT"END OF PROGRAM."
1860 END
```

Variables and Strings

A$ - General string input
N$ - First name of spouse, girlfriend, or boyfriend
C$(I) - Subscripted, for up to 10 complaints
P(I) - Subscripted, for up to 10 priority numbers
C - Counter, number of complaints
I - Total of variable C minus 1
C$ and X - Used for placing complaints in order of priority
WW - Will be equal to 1 when both parties have entered complaints
NN$ - Current user (entering data)
W$(I) and W(I) - Take on the contents of C$(I) and P(I) before the second user enters data
X$ - Contents of NN$ (first name)
F - Value of I (total, first users entries)
G - Value of I (total, second users entries)
R - For correct message output
JJ - Correct message output

Explanation of the Program Lines

Line 20 clears 1000 bytes of memory for string storage.

Lines 40-170 ask the first user for the first name of his or her spouse, girlfriend, or boyfriend. If the questions are not applicable, the program is terminated.

Lines 180-310 display the general instructions.

Lines 320-460 lets the current user input up to 10 complaints.

Line 410 sends program control to line 1220, which tests the length of each complaint. This insures that the printout (chart) will be kept as neat and readable as possible.

Lines 470-480 sets the value of I to C-1. If I= 0 (zero), the program will loop back to line 370, the beginning of the complaint entries.

Lines 490-640 allow the user to input priority values for each complaint entered.

Lines 650-780 rearrange the complaints according to their priority numbers, so that the complaints are arranged from highest priority to lowest priority when printed.

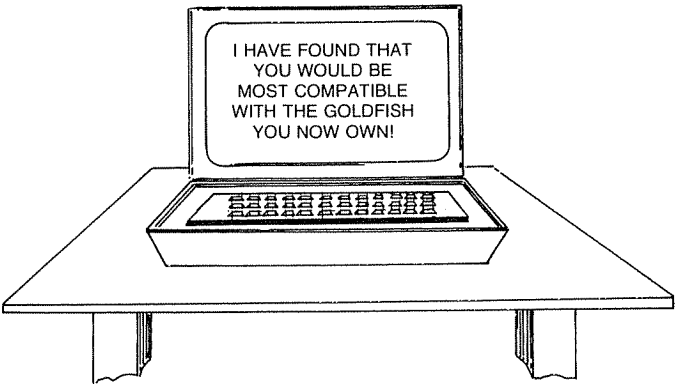
Lines 790-1000 allow the users partner to register his or her complaints. Lines 930-980 transfer the current data to alternate strings and variables, so the information will not be lost when the second person enters his or her complaints. Line 1000 sends program control back to line 330 for the entry of the new complaints.

Lines 1010-1190 display each user's complaints and priorities in order from high (most important) to low (least important).

Lines 1310-1440 display generalized messages for each user.

Lines 1450-1730 display each user's complaints one at a time, and allow them to discuss each one before continuing.

Lines 1750-1860 display the final message and end the program.



DATE SERVICE

Been having problems finding the right person to date? Have no fear; your computer will now try and rescue you! Naturally, you and up to 9 others (of the opposite sex) will answer all necessary questions. After this, the computer will compile all data collected and tell you who you would be most compatible with. Keep in mind that this is only a simulation or game. The computer has no way of judging the final outcome of a person's thought pattern.

Sample Run

```
**DATE SERVICE**

OPERATING INSTRUCTIONS? YES

BEEN HAVING PROBLEMS FINDING
THE RIGHT PERSON TO DATE?
HAVE NO FEAR - COMPUTER DATE
SERVICE IS HERE !!
NATURALLY, YOU'LL NEED TO INPUT
REQUIRED DATA SO THE COMPUTER
CAN SORT AND SELECT A PERSON
THAT MIGHT BE COMPATIBLE WITH YOU.

YOUR FIRST NAME? MAC
AGE? 31
MALE OR FEMALE (1 or 2)? 1
```

USING ONE WORD FOR EACH, ANSWER
THESE QUESTIONS?

YOUR HOBBY?COMPUTERS
YOUR INTEREST?MONEY
WHAT YOU LIKE?FOOD
WHAT YOU DISLIKE?NOISE
TYPE OF MUSIC? EASY
TYPE OF CAR? LARGE
COLOR CHOICE? BLUE

THE NEXT PART OF THE PROGRAM
WILL BE DEALT WITH IN ONE
OF TWO WAYS:
FIRST, YOU CAN HAVE SOMEONE
ELSE ANSWER THE QUESTIONS.
SECOND, YOU CAN ANSWER THEM
ABOUT PERSONS YOU KNOW.

WHAT WILL IT BE (1 or 2)? 1

SEND IN FEMALE # 1
TO ANSWER THE QUESTIONS. YOU
SHOULD NOT BE PRESENT, MAC.

PRESS ENTER WHEN READY

YOUR FIRST NAME? SARAH
AGE? 29
USING ONE WORD FOR EACH, ANSWER
THESE QUESTIONS?

YOUR HOBBY?COOKING
YOUR INTEREST?SEWING
WHAT YOU LIKE? PARTIES
WHAT YOU DISLIKE?NOISE
TYPE OF MUSIC? EASY
TYPE OF CAR? SMALL
COLOR CHOICE? RED

SEND IN FEMALE # 2

TO ANSWER THE QUESTIONS. YOU
SHOULD NOT BE PRESENT, SARAH.

PRESS ENTER WHEN READY

YOUR FIRST NAME? SANDRA
AGE? 32
USING ONE WORD FOR EACH, ANSWER
THESE QUESTIONS?

YOUR HOBBY?COMPUTERS
YOUR INTEREST?COLLECTING
WHAT YOU LIKE?FOOD
WHAT YOU DISLIKE?BUGS
TYPE OF MUSIC? ROCK
TYPE OF CAR? SMALL
COLOR CHOICE? BLUE

SEND IN FEMALE # 3
TO ANSWER THE QUESTIONS, YOU
SHOULD NOT BE PRESENT, SANDRA.

PRESS ENTER WHEN READY

YOUR FIRST NAME? MARTHA
AGE? 28
USING ONE WORD FOR EACH, ANSWER
THESE QUESTIONS?

YOUR HOBBY?MEN
YOUR INTEREST?MONEY
WHAT YOU LIKE?DIAMONDS
WHAT YOU DISLIKE?ANIMALS
TYPE OF MUSIC? EASY
TYPE OF CAR? LARGE
COLOR CHOICE? GREEN

SEND IN FEMALE # 4
TO ANSWER THE QUESTIONS. YOU
SHOULD NOT BE PRESENT, MARTHA.

PRESS ENTER WHEN READY

YOUR FIRST NAME? ROBIN
AGE? 35
USING ONE WORD FOR EACH, ANSWER
THESE QUESTIONS?

YOUR HOBBY?STAMPS
YOUR INTEREST?FLYING
WHAT YOU LIKE?LARGE HOUSES
WHAT DO YOU DISLIKE? SMALL HOUSES
TYPE OF MUSIC? ANY KIND
TYPE OF CAR? LARGE
COLOR CHOICE? BLUE

SEND IN FEMALE # 5
TO ANSWER THE QUESTIONS. YOU
SHOULD NOT BE PRESENT, ROBIN.

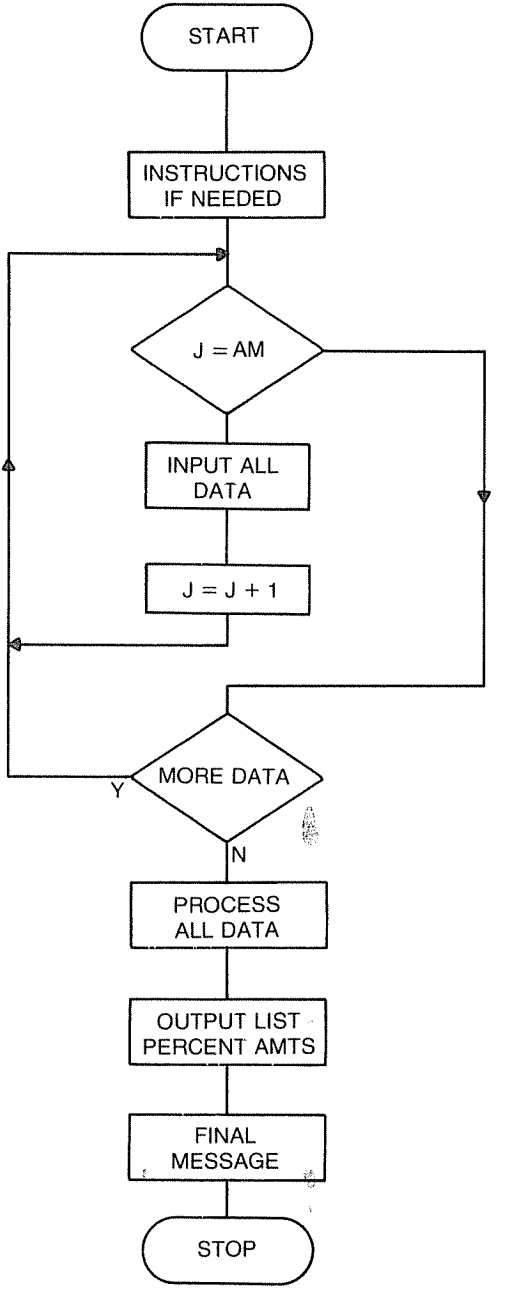
PRESS ENTER WHEN READY

YOUR FIRST NAME? SHERRY
AGE? 27
USING ONE WORD FOR EACH, ANSWER
THESE QUESTIONS?

YOUR HOBBY?FURNITURE
YOUR INTEREST?CARS
WHAT YOU LIKE?MONEY
WHAT YOU DISLIKE?FAST FOODS
TYPE OF MUSIC? COUNTRY
TYPE OF CAR? SMALL
COLOR CHOICE? PURPLE

IF MAC IS NOT PRESENT, SEND
HIM TO THE COMPUTER, PLEASE.

MAC, 5 WOMEN ANSWERED THE QUESTIONNAIRE.
DO YOU HAVE MORE, OR SHALL
I PROCESS THE CURRENT DATA.
(MORE OR PROCESS)? PROCESS
I WILL NOW SORT AND PROCESS THE DATA
YOU WILL KNOW IF ONE WILL BE
COMPATIBLE IN A MOMENT MAC.



Flowchart for Date Service.

```
NAME                PERCENT COMPATIBLE

SANDRA.....          50 %
MARTHA.....          50 %
SARAH.....           37.5 %
ROBIN.....           37.5 %
SHERRY.....          12.5 %

ENTER?

REMEMBER, THIS IS ONLY A
COMPUTER PROGRAM
EXPRESSING WHO MIGHT BE
COMPATIBLE WITH YOU THROUGH
ENTERED DATA....

END OF**DATE SERVICE**
```

Program Listing

```
 S10 REM PROGRAM TITLE: DATE SERVICE
 20 CLEAR 1000:DIM R(12)
 30 RANDOM:J=0.AM=5
 40 CLS:M$="MALE":F$="FEMALE":U$="YOUR"
 50 PRINT"** DATE SERVICE **"
 60 PRINT:E$="YOU"
 70 PRINT"OPERATING INSTRUCTIONS (Y/N)";
 80 INPUT A$
 90 IF LEFT$(A$,1)<>"Y" THEN 190
100 PRINT
110 PRINT"BEEN HAVING PROBLEMS FINDING"
120 PRINT"THE RIGHT PERSON TO DATE ?"
130 PRINT"HAVE NO FEAR - COMPUTER DATE"
140 PRINT"SERVICE IS HERE !!"
150 PRINT"NATURALLY, YOU'LL NEED TO INPUT"
160 PRINT"REQUIRED DATA SO THE COMPUTER"
170 PRINT"CAN SORT AND SELECT A PERSON"
180 PRINT"THAT MIGHT BE COMPATIBLE WITH YOU."
190 PRINT
200 PRINT U$;" FIRST NAME";
205 INPUT N$(J):IF N$(J)="" THEN 200
210 INPUT"AGE";A(J)
215 IF J<>0 THEN PRINT:GOTO 230
220 INPUT"MALE OR FEMALE (1 OR 2)";S:PRINT
225 IF S=1 THEN W$=F$ ELSE W$=M$
230 PRINT"USING ONE WORD FOR EACH, ANSWER"
```

```
240 PRINT"THESE QUESTIONS  ";
245 IF SE<>2 THEN PRINT N$(J);"'S" ELSE PRINT
250 PRINT U$;" HOBBY";:INPUT A$(J)
260 PRINT U$;" INTEREST";:INPUT B$(J)
270 PRINT'WHAT ";E$;" LIKE";
272 IF SE=2 PRINT"S";
275 INPUT C$(J)
280 PRINT"WHAT ";E$;" DISLIKE";
282 IF SE=2 PRINT"S";
285 INPUT D$(J)
290 INPUT"TYPE OF MUSIC";E$(J)
300 INPUT"TYPE OF CAR";F$(J)
310 INPUT"COLOR CHOICE";G$(J)
320 CLS:IF J=AM THEN 620
325 IF J<>0 PRINT:J=J+1:GOTO 430
330 PRINT"THE NEXT PART OF THE PROGRAM"
340 PRINT"WILL BE DEALT WITH IN ONE"
350 PRINT"OF TWO WAYS:"
360 PRINT"FIRST, YOU CAN HAVE SOMEONE"
370 PRINT"ELSE ANSWER THE QUESTIONS."
380 PRINT"SECOND, YOU CAN ANSWER THEM"
390 PRINT"ABOUT PERSONS YOU KNOW."
400 PRINT:J=J+1
410 PRINT"WHAT WILL IT BE (1 OR 2)";
420 INPUT SE:N$=N$(0)
430 IF SE=1 THEN 460
440 IF SE=2 THEN 540
450 GOTO 400
460 REM OTHERS TO INPUT
470 PRINT"SEND IN ";W$;" #";J
480 PRINT"TO ANSWER THE QUESTIONS, YOU"
490 PRINT"SHOULD NOT BE PRESENT, ";
495 IF J=6 PRINT:GOTO 510
500 PRINT N$(J-1);" ":PRINT
510 PRINT"PRESS ENTER WHEN READY";
520 INPUT X$
530 CLS:GOTO 190
540 REM USER TO INPUT
545 IF J>=2 PRINT"ANSWER ";:GOTO 600
550 PRINT"YOU ARE ABOUT TO ANSWER THE"
560 PRINT"QUESTIONS FOR THE ";W$;"'S"
570 PRINT"THAT YOU KNOW.  DO NOT BE"
580 PRINT"ONE SIDED TO TRY AND OBTAIN"
590 PRINT"A COMPATIBLE PERSON.  ANSWER"
600 PRINT"AS FACTUALLY AS POSSIBLE, ";
610 PRINT N$;" ":GOSUB 850:PRINT:GOTO 510
620 REM DATA IN
630 CLS
```

```
640 PRINT"IF ";N$;" IS NOT PRESENT, SEND"
650 IF S=1 PRINT"HIM ";:GOTO 670
660 PRINT"HER ";
670 PRINT"TO THE COMPUTER, PLEASE."
680 PRINT:IF AA=AM-X THEN PRINT:GOTO 890
685 IF SE=2 PRINT"YOU HAVE ";:GOTO 720
690 PRINT N$;",";J;
700 IF S=1 PRINT"WOMEN ";:GOTO 720
710 PRINT"MEN ";
720 PRINT"ANSWERED THE QUESTIONNAIRE."
730 PRINT"DO YOU HAVE MORE, OR SHALL"
740 PRINT"I PROCESS THE CURRENT DATA."
750 INPUT"(MORE OR PROCESS)";Q$
760 IF Q$="MORE" THEN 790
770 IF Q$="PROCESS" THEN PRINT:GOTO 890
780 PRINT:GOTO 690
790 PRINT N$;" UP TO";10-J;"MORE CAN"
800 PRINT"ANSWER, THEN I MUST PROCESS."
810 INPUT"HOW MANY MORE DO YOU HAVE";X
820 AA=AM:AM=AM+X
830 IF AM>=11 THEN AM=AA:PRINT:GOTO 790
840 J=J+1:PRINT:IF SE=2 THEN 510 ELSE 460
850 REM FOR INPUT
855 IF J>=2 RETURN
860 IF S=1 THEN U$="HER":E$="SHE":GOTO 880
870 U$="HIS":E$="HE"
880 RETURN
890 IF AA<AM THEN AR=AM ELSE AR=AA
900 PRINT"I WILL NOW SORT AND PROCESS DATA"
910 PRINT"YOU WILL KNOW IF ONE WILL BE"
920 PRINT"COMPATIBLE IN A MOMENT ";N$;"."
930 I=0:X=1:P=1
940 IF ABS(A(I)-A(X))<=5 THEN GOSUB 1150
950 ON P GOTO 960,970,980,990,1000,1010,1020
960 T$=A$(I):A$=A$(X):GOTO 1030
970 T$=B$(I):A$=B$(X):GOTO 1030
980 T$=C$(I):A$=C$(X):GOTO 1030
990 T$=D$(I):A$=D$(X):GOTO 1030
1000 T$=E$(I):A$=E$(X):GOTO 1030
1010 T$=F$(I):A$=F$(X):GOTO 1030
1020 T$=G$(I):A$=G$(X)
1030 IF T$<>A$ THEN 1050
1040 GOSUB 1150:REM MATCH
1050 P=P+1
1060 IF P=8 THEN 1080
1070 GOTO 950
1080 R(X)=TR:P=1
1090 X=X+1
```

```
1100 IF X>AR THEN 1180
1110 TR=0:GOTO 940
1150 REM POINT PER MATCH
1160 TR=TR+12.5
1170 RETURN
1180 REM PERCENT
1190 I=1:F=0
1200 IF R(I)=0 THEN 1240
1210 IF R(I)>=R(I+1) THEN 1240
1220 R=R(I):R(I)=R(I+1):R(I+1)=R:F=1
1230 N$=N$(I):N$(I)=N$(I+1):N$(I+1)=N$
1240 I=I+1
1250 IF I>AR THEN 1270
1260 GOTO 1200
1270 IF F=1 THEN 1190
1280 REM LIST
1290 PRINT"NAME" TAB(20)"PERCENT COMPATIBLE"
1300 PRINT:FOR I=1 TO AR
1310 IF R(I)=0 THEN 1330
1320 PRINT N$(I);"......"TAB(25) R(I);"%"
1330 NEXT
1340 PRINT:FOR I=1 TO 2000
1350 NEXT:INPUT"ENTER";X:CLS
1360 PRINT"REMEMBER, THIS IS ONLY A"
1370 PRINT"COMPUTER  PROGRAM"
1380 PRINT"EXPRESSING WHO MIGHT BE"
1390 PRINT"COMPATIBLE WITH YOU THROUGH"
1400 PRINT"ENTERED DATA..."
1410 PRINT
1420 PRINT"END OF ** DATE SERVICE **"
1430 END
```

Variables and Strings

J - Counter, total persons entered
AM - Current amount
M$,F$,U$, & E$ - Are used for print messages, depending upon the sex of the main user
N$(J) - Name of person
A(J) - Age of user
S & SE - Are used for print messages again, depending upon the sex of the main user
A$(J) & G$(J) - Are used to enter necessary data
AA - Value of AM, if additional persons are to be included
P - Counter, for comparing entries
R(X) - Total points, when entries match
TR - Number of points per match

Explanation of the Program Lines

Line 30 sets two variables: one is a counter; the other is the total number of potential partners that can respond during the initial section of the program.

Lines 40-60 contain strings; the contents of these will change according to the questions and the sex of the user.

Lines 70-180 display the general operating instructions.

Lines 200-310 form the main body of the program. The main user answers all the questions first.

Lines 320-440 begin the second part of the program, if J<>AM.

Lines 460-530 call the other users to answer all necessary questions.

Lines 540-610 are used for the main (first) user if he or she has chosen to answer the questions about people of the opposite sex. If the main user decides to answer these questions, he or she should not be one-sided as this will defeat the purpose of the program.

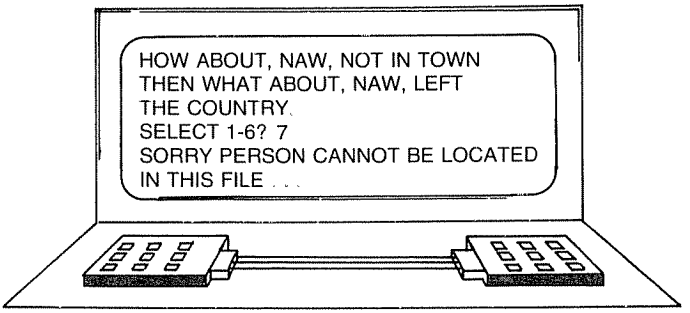
Lines 620-750 call the main user back to the computer. At this point, he or she will be asked if there are more persons to input data.

Line 760 branches to lines 790-840 for data entries involving up to 5 more people.

Line 770 jumps to line 890 for the processing of the data entered.

Lines 890-1270 compare each piece of data to the main users data. If a match is made 12.5 points are added to the total score. No points are given if the data does not match.

Lines 1280-1430 print the entire list of names. The percentage of compatability is also printed if it is not equal to 0.



KEY A DATE

No the computer will not not run out and get you a date! You'll be able to select more easily who you want to date though. With the help of this program, you'll build yourself a file for your own personal use that no one else can pry into. If another person comes along to nose into your file, they will be surprised if they don't enter the correct code and the entire program disappears!

Sample Run

```
><><KEY A DATE><><

1) INSTRUCTIONS
2) SEND DATA / RETRIEVE DATA
3) SEE A NUMBER / ADD A NAME
4) DELETE A NAME
5) SEE ENTIRE LIST
6) TERMINATE RUN
 1

THE INSTRUCTIONS ARE QUITE STRAIGHT-
FORWARD AND SIMPLE. ALMOST ALL
OF US THAT ARE NOT MARRIED (AND
SOME THAT ARE) CARRY AROUND A
LITTLE BOOK. THIS LITTLE BOOK
CONTAINS NAMES AND PHONE NUMBERS
OF PERSONS OF THE OPPOSITE SEX
WITH WHOM YOU MIGHT WANT TO
MAKE A DATE. NOW YOUR COMPUTER
IS ABOUT TO RELIEVE YOU OF THE
```

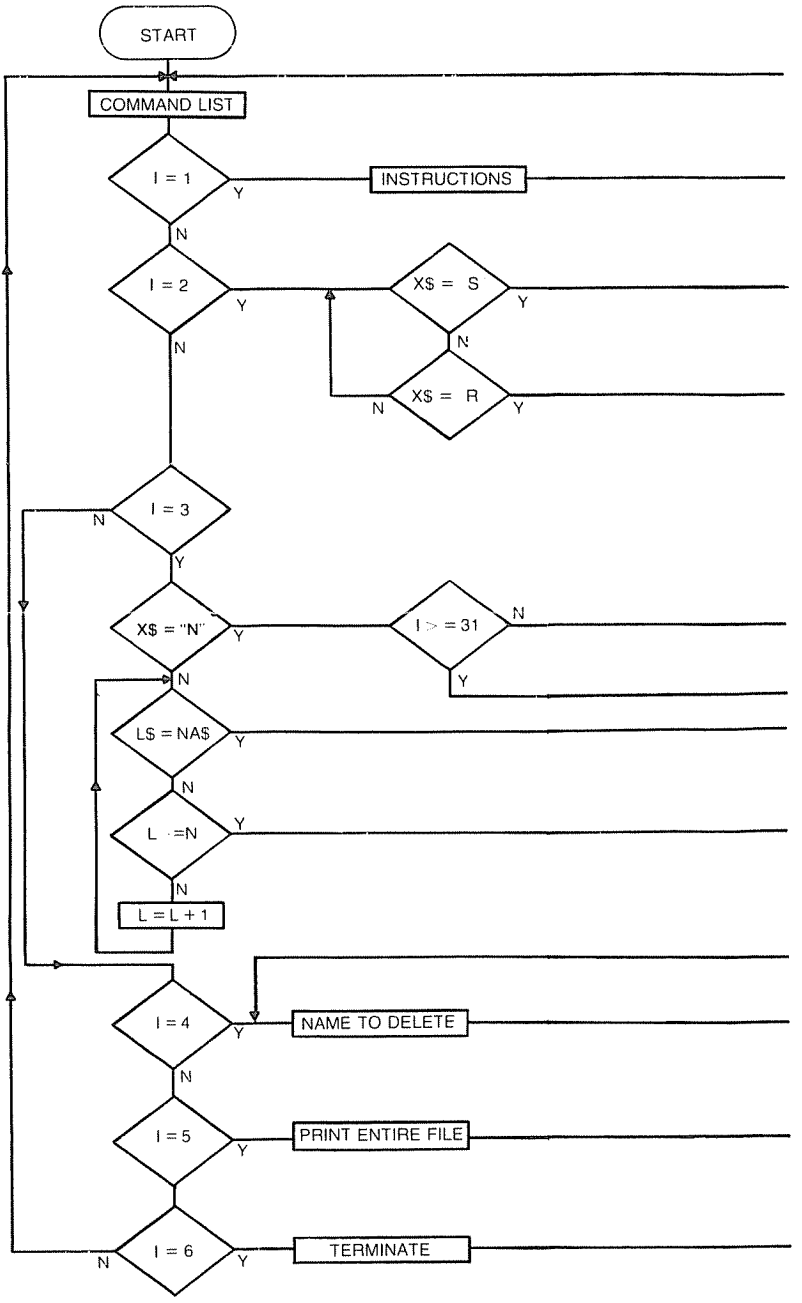
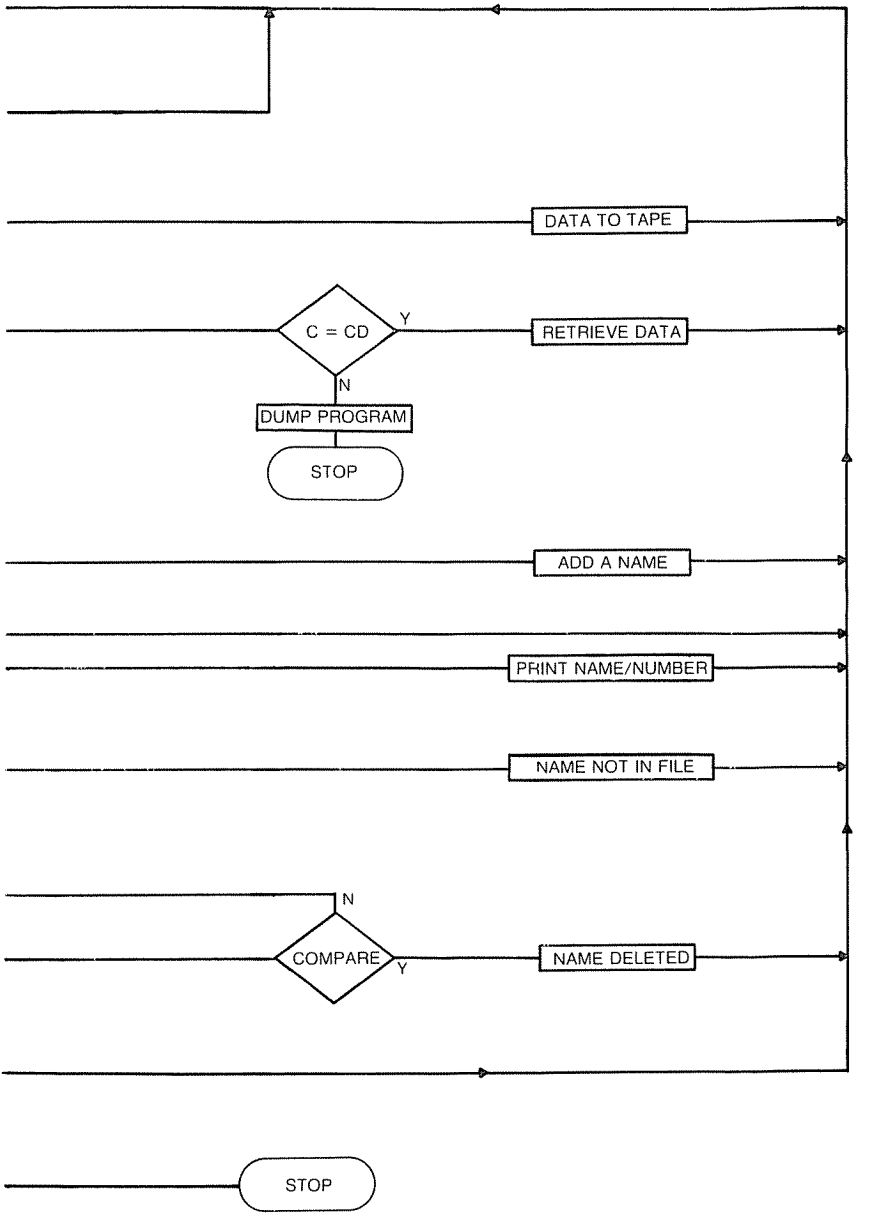


Fig. 3-14. Flowchart for Key a Date.


DATA TO TAPE
C = CD
Y
RETRIEVE DATA
N
DUMP PROGRAM
STOP
ADD A NAME
PRINT NAME/NUMBER
NAME NOT IN FILE
N
COMPARE
Y
NAME DELETED
STOP

CHORE OF THUMBING THROUGH
THAT LITTLE BOOK AND GIVE ONLY
YOU ACCESS TO ITS CONTENTS.

STRIKE A KEY
YOU WILL BE ALLOWED TO ENTER
UP TO 30 NAMES AND PHONE NUMBERS
OF THE PEOPLE THAT YOU CHOOSE
ALONG WITH THIS YOU WILL ENTER
A CODE NUMBER (5 DIGIT). THIS
CODE WILL BE SAVED ON TAPE
ALONG WITH ALL THE OTHER DATA.
WHEN YOU WANT TO RETRIEVE THE
DATA, YOU'LL NEED TO REMEMBER
YOUR CODE...OR YOU'LL NEVER
SEE THE DATA AGAIN !!!

STRIKE A KEY
><>< KEY A DATE ><><

1) INSTRUCTIONS
2) SEND DATA / RETRIEVE DATA
3) SEE A NUMBER / ADD A NAME
4) DELETE A NAME
5) SEE ENTIRE LIST
6) TERMINATE RUN

Program Listing

```
 10 REM PROGRAM TITLE: KEY A DATE
 20 CLEAR 1200:CLS
 30 DIM N$(30),P$(30):N=1
 40 PRINT")<)< KEY A DATE )<)<"
 50 PRINT
 60 PRINT"1) INSTRUCTIONS"
 70 PRINT"2) SEND DATA / RETRIEVE DATA"
 80 PRINT"3) SEE A NUMBER / ADD A NAME"
 90 PRINT"4) DELETE A NAME"
100 PRINT"5) SEE ENTIRE LIST"
110 PRINT"6) TERMINATE RUN"
120 INPUT I
130 ON I GOTO 140,460,690,910,1310,1600
140 CLS
150 PRINT"THE INSTRUCTIONS ARE QUITE
     STRAIGHT-"
```

```
160 PRINT"FORWARD AND SIMPLE. ALMOST ALL"
170 PRINT"OF US THAT ARE NOT MARRIED (AND"
180 PRINT"SOME THAT ARE) CARRY AROUND A"
190 PRINT"LITTLE BOOK. THIS LITTLE BOOK"
200 PRINT"CONTAINS NAMES AND PHONE NUMBERS"
210 PRINT"OF PERSONS OF THE OPPOSITE SEX"
220 PRINT"WITH WHOM YOU MIGHT WANT TO"
230 PRINT"MAKE A DATE WITH. WELL NOW YOUR"
240 PRINT"COMPUTER IS ABOUT TO RELIEVE YOU"
250 PRINT"OF THE CHORE OF THUMBING THROUGH"
260 PRINT"THAT LITTLE BOOK AND GIVE ONLY"
270 PRINT"YOU ACCESS TO ITS CONTENTS."
280 PRINT:GOSUB 290:GOTO320
290 PRINT"STRIKE A KEY"
300 X$=INKEY$:IF X$="" THEN 300
310 CLS:RETURN
320 PRINT"YOU WILL BE ALLOWED TO ENTER"
330 PRINT"UP TO 30 NAMES AND PHONE NUMBERS"
340 PRINT"OF THE PEOPLE THAT YOU CHOOSE."
350 PRINT"ALONG WITH THIS YOU WILL ENTER"
360 PRINT"A CODE NUMBER (5 DIGIT). THIS"
370 PRINT"CODE WILL BE PROCESSED TO TAPE"
380 PRINT"ALONG WITH ALL THE OTHER DATA."
390 PRINT"WHEN YOU WANT TO RETRIEVE THE"
400 PRINT"DATA, YOU'LL NEED TO REMEMBER"
410 PRINT"YOUR CODE. . OR YOU'LL NEVER"
420 PRINT"SEE THE DATA AGAIN !!!"
425 ' IF PROGRAM IS LEFT AS-IS
430 PRINT
440 GOSUB 290
450 GOTO 40
460 ' SEND / RETRIVE DATA
470 CLS
480 INPUT"SEND OR RETRIEVE (S/R)";X$
490 IF X$="S" THEN 520
500 IF X$="R" THEN 590
510 PRINT:GOTO 480
520 CLS
530 PRINT"READY PLAYER (IN RECORD)"
540 INPUT"THEN KEY ENTER";X$
550 PRINT #-1,CD:PRINT #-1,N
560 FOR I=1 TO N
570 PRINT #-1,N$(I),P$(I):NEXT
580 CLS:GOTO 40
590 ' RETRIEVE DATA
600 CLS
610 PRINT"READY PLAYER (IN PLAY)"
620 INPUT"THEN ENTER CODE";C
```

```
630 INPUT #-1,CD
640 IF C<>CD THEN NEW
650 INPUT #-1,N
660 FOR I=1 TO N
670 INPUT #-1,N$(I),P$(I):NEXT
680 CLS:GOTO 40
690 ' ADD A NAME
700 ' SEE A NUMBER
710 CLS:GOTO 1100
720 PRINT"** ADD A NAME **"
730 IF N<>1 THEN 840
740 PRINT"INPUT FULL NAME, THEN PHONE"
750 PRINT"NUMBER. USE A - TO SEPARATE"
760 PRINT"PARTS OF THE NUMBER."
770 PRINT"TO CANCEL, PRESS 'ENTER' ONLY"
780 PRINT"FOR A NAME."
790 PRINT
800 INPUT N$(N)
810 IF N$(N)="" THEN PRINT:GOTO 855
820 INPUT P$(N)
830 N=N+1
840 IF N>=31 THEN N=N-1:GOTO 870
850 GOTO 790
855 IF CD<>0 THEN 865
860 IF N<>1 THEN INPUT"5 DIGIT CODE";CD
865 GOSUB 290:CLS:GOTO 40
870 PRINT
880 PRINT"YOU HAVE ENTERED THE LIMIT"
890 PRINT"FOR THE PROGRAM,";N;"NAMES."
900 PRINT:IF PR=0 GOTO 855 ELSE 865
910 ' DELETE A NAME
920 CLS:PRINT"** DELETE A NAME **"
930 PRINT"TO DELETE A NAME AND NUMBER,"
940 PRINT"INPUT THE ENTIRE NAME OF THE"
950 PRINT"PERSON, AS IT APPEARS IN THE"
960 PRINT"FILE. WHO IS THE PERSON YOU"
970 INPUT"WANT TO REMOVE FROM LIST";N$
980 FOR I=1 TO N
990 IF N$=N$(I) THEN 1060
1000 NEXT.PRINT
1010 IF P=0 THEN 1040
1020 PRINT N$;" REMOVED FROM FILE."
1030 P=0:GOTO 900
1040 PRINT N$;" COULD NOT BE LOCATED."
1050 GOTO 1030
1060 N$(I)="":P$(I)=""
1070 P=1:GOSUB 1460:IF N=0 N=1
1080 PRINT:GOTO 1010
1100 ' SEE A NAME?
```

```
1110 INPUT"SEE A NUMBER (Y/N)";X$
1120 IF X$="N" THEN CLS:GOTO 720
1130 PRINT:IF N=1 AND N$(N)="" THEN 1550
1140 PRINT"INPUT PERSON'S LAST NAME ONLY";
1150 INPUT L$
1160 L=1
1170 LE=LEN(N$(L))
1180 FOR I=LE TO 1 STEP-1:IF I=0 THEN 1280
1185 IF MID$(N$(L),I,1)=" " AND D=0 THEN 1420
1190 NA$=MID$(N$(L),I,1)+NA$
1200 NEXT
1210 PRINT NA$ ;
1220 INPUT" (Y/N)";X$
1230 IF X$="Y" THEN 1260
1240 IF X$="N" THEN 1280
1250 PRINT:GOTO 1210
1260 PRINT"THE NUMBER:   ";P$(L):NA$=""
1270 D=0:PRINT:GOSUB 290:GOTO 40
1280 NA$="":L=L+1
1290 IF L>=N THEN N$=L$:GOTO 1040
1300 GOTO 1170
1310 ' ENTIRE FILE
1320 L=1:A=1:CLS
1330 PRINT"NAME / NUMBER"
1340 PRINT N$(L),P$(L)
1350 L=L+1:A=A+1
1360 IF L>=N THEN 1390
1370 IF A>=8 THEN A=1:PRINT:GOSUB 290:L=L-1
1380 GOTO 1340
1390 PRINT
1400 GOSUB 290
1410 GOTO 40
1420 ' RIGHT NAME
1430 IF NA$<>L$ THEN 1280
1440 D=1
1450 GOTO 1190
1460 ' RE-SORT FILE
1470 FOR I=1 TO N
1480 IF N$(I)="" THEN 1500
1490 NEXT:RETURN
1500 J=I+1
1510 N$(I)=N$(J):P$(I)=P$(J)
1520 I=I+1:J=J+1
1530 IF J>N THEN N=N-1:RETURN
1540 GOTO 1510
1550 ' NONE IN FILE
1560 PR=1
1570 PRINT"** NO NAMES IN FILE **"
1580 GOTO 900
```

```
1600  ' TERMINATE
1610  PRINT
1620  PRINT"END OF PROGRAM"
1630  END
```

Variables and Strings

N$(I) - Names in the file
P$(I) - Phone numbers in the file
N - Counter, number in file
X$ - General input string
CD - 5 digit code number
P - Either equal to 0 or 1 for deleting a name
D - Either equal to 0 or 1 for searching for a particular name
L & A - Counters for printing all names in the current list
PR - Name not in file (if equal to 1)

Explanation of the Program Lines

Line 20 clears 1200 bytes of memory for string storage.

Line 30 dimensions the N$ and P$, arrays for the names and phone numbers.

Lines 40-120 display the command list.

Line 130 sends control to the appropriate line depending on the value input for I.

Lines 140-440 display the general instructions.

Lines 460-680 either send data to the tape player or retrieve it from the tape player. If the program is left as-is the user must enter the 5 digit code that was placed on the tape in order to retrieve data. If the user fails to enter the correct code, C< >CD, the entire program in memory will be 'dumped'.

Lines 690-900 and Lines 1100-1300 let the user either see a phone number or add a name. With the program as-is, up to 30 names and phone numbers can be entered.

Lines 910-1080 allow the user to delete a name and number from the current file.

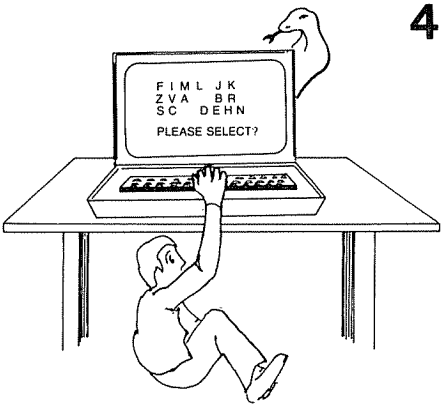
Lines 1310-1410 display the entire file that is currently in memory.

Lines 1420-1450 test for a correct name when the user wants to locate a certain number only last names are used.

Lines 1460-1540 re-sorts the entire file when a name has been deleted.

Lines 1550-1580 display a message if there are no names in the file.

Lines 1600-1630 terminate the program.



Games That Span the Ages

Squares is a game like checkers, but you can move in any direction you want. Challenge a friend or the computer.

Hold Time allows you to stop the computer's piece while you shoot it. It's not that easy though: your lasers must be at the right location at the right moment.

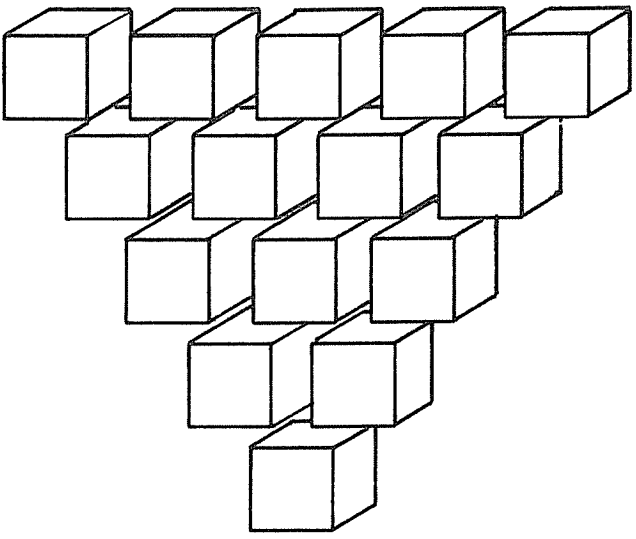
Dangerous Affair challenges even the best would-be spy by making you work with coded messages and incompetent agents.

Pennies or Dollars offers you the chance to get rich quick or to become poor in just as short a time.

Be Prepared (Part I) tests your dexterity and your luck. You'll need both to successfully dock your spaceship.

Rain of Terror is a test of your courage. How long do you dare remain at the spot where the bomb will hit?

Knights places you in medieval times. The king is depending on you to destroy the outlaws without killing their hostages.

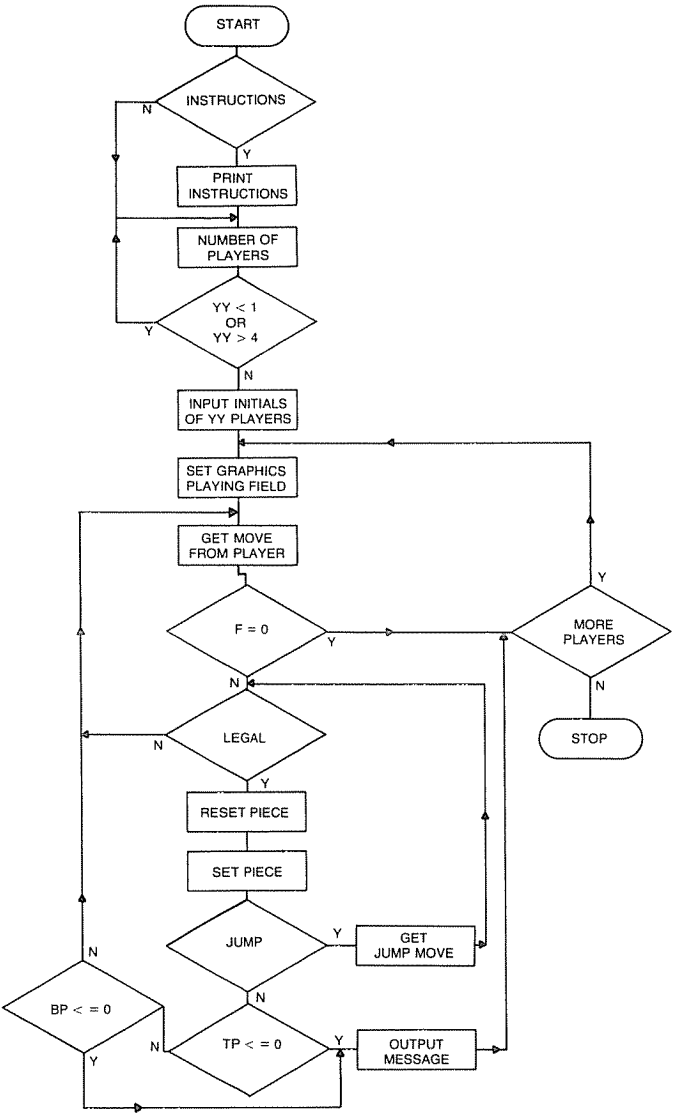


SQUARES

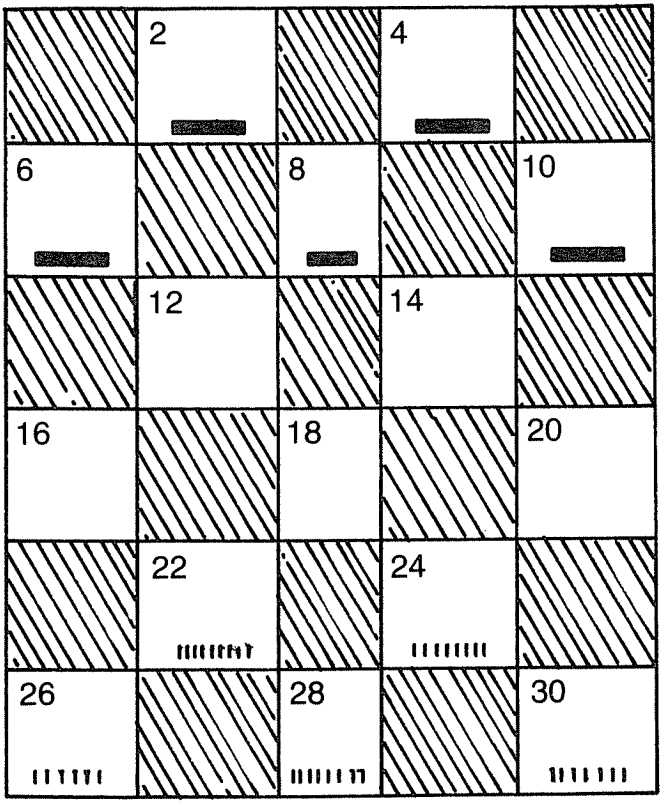
This program will give you all of the action that checkers never attempted to have! Without having to get "crowned" you can move wherever you wish on the playing board, just so long as you do it to win! If 4 people are playing, the victor of the first game can play the victor of the second game. Of course, you can play by yourself against the computer.

Sample Run

```
SEE INSTRUCTIONS? YES

S Q U A R E S

HOW MANY TO PLAY (1-4)? 2
JUST LIKE CHECKERS, YOU WILL MOVE
FROM SQUARE TO SQUARE TRYING TO
OUTWIT YOUR OPPONENT AND JUMP
THEIR PIECE. BUT, UNLIKE CHECKERS,
EACH WILL ONLY HAVE 5 PLAYING
PIECES. YOU CAN MOVE FORWARD OR
BACKWARD, AS IF ALL PIECES ARE
KINGS. THE VICTOR WILL BE THE
PLAYER WITH PLAYING PIECES
```



Flowchart for *Squares*.



REMAINING. TO THROW IN THE TOWEL
(GIVE UP) JUST ENTER A 0 (ZERO) WHEN
YOU ARE ASKED 'FROM?'

PRESS ENTER?

IF YOU ARE PLAYING BY YOURSELF,
YOU WILL PLAY AGAINST THE COMPUTER.
IF THREE ARE PLAYING, ONE WILL
BE CHOSEN TO PLAY THE WINNER OF
THE FIRST ROUND. IF FOUR ARE PLAYING
THE COMPUTER WILL CHOOSE OPPONENTS
AND THE WINNER OF ONE GAME CAN
PLAY THE WINNER OF THE FOLLOWING

GAME. REMEMBER, MOVE ONE SQUARE AT A TIME UNLESS YOU ARE JUMPING.

PRESS ENTER TO START?

I NEED THE INITIALS OF THE 2 PLAYERS? AM
?? AJ

Program Listing

```
 10 REM PROGRAM TITLE: SQUARES
 20 CLS:RANDOM
 30 CLEAR 500
 40 DIM X(40),Y(40),W(40)
 50 QQ=-99
 60 FF=1
 70 INPUT"SEE INSTRUCTIONS (Y/N)";A$
 80 IF A$="Y" OR A$="YES" THEN 120
 90 IF A$="N" OR A$="NO" THEN TN=1:GOTO 150
100 PRINT"TRY A SOLID ANSWER..."
110 PRINT:GOTO 70
120 PRINT CHR$(143);CHR$(143);
130 PRINT" S Q U A R E S ";CHR$(143);
140 PRINT CHR$(143)
150 PRINT
160 PRINT"HOW MANY TO PLAY (1-4)";
170 INPUT YY
180 IF YY<1 OR YY>4 THEN 150
190 IF YY=1 THEN GL=1
200 REM INSTRUCTIONS NOT NEEDED
210 IF TN=1 THEN PRINT:GOTO 510
220 REM INSTRUCTIONS
240 PRINT"JUST LIKE CHECKERS, YOU WILL MOVE"
250 PRINT"FROM SQUARE TO SQUARE TRYING TO"
260 PRINT"OUTWIT YOUR OPPONENT AND JUMP"
270 PRINT"THEIR PIECE. BUT, UNLIKE CHECKERS,"
280 PRINT"EACH WILL ONLY HAVE 5 PLAYING"
290 PRINT"PIECES. YOU CAN MOVE FORWARD OR"
300 PRINT"BACKWARD, AS IF ALL PIECES ARE"
310 PRINT"KINGS. THE VICTOR WILL BE THE"
320 PRINT"PLAYER WITH PLAYING PIECES"
330 PRINT"REMAINING. TO THROW IN THE TOWEL"
340 PRINT"(GIVE UP) JUST ENTER A 0
    (ZERO) WHEN"
350 PRINT"YOU ARE ASKED 'FROM ?'"
```

```
360 PRINT
370 INPUT"PRESS ENTER";XX
380 CLS
390 PRINT"IF YOU ARE PLAYING BY YOURSELF,"
400 PRINT"YOU WILL PLAY AGAINST THE COMPUTER."
410 PRINT"IF THREE ARE PLAYING, ONE WILL"
420 PRINT"BE CHOSEN TO PLAY THE WINNER OF"
430 PRINT"THE FIRST ROUND.IF FOUR ARE PLAYING"
440 PRINT"THE COMPUTER WILL CHOOSE OPPONENTS"
450 PRINT"AND THE WINNER OF ONE GAME CAN"
460 PRINT"PLAY THE WINNER OF THE FOLLOWING"
470 PRINT"GAME.REMEMBER,MOVE ONE SQUARE AT A"
480 PRINT"TIME UNLESS YOU ARE JUMPING."
490 PRINT
500 INPUT"PRESS ENTER TO START";XX
510 CLS
520 IF TN=1 AND YY=1 THEN 540
530 IF YY>1 GOTO 750
540 AA=RND(2)
550 IF AA=1 THEN RR=1:GOTO 570
560 RR=3:GOTO 600
570 PRINT"YOU WILL PLAY FIRST..."
580 PRINT"YOU HAVE THE SOLID PIECES."
590 PRINT:GOTO 610
600 PRINT"I WILL PLAY FIRST..."
610 PRINT:IF Z<>0 THEN 1395
750 MZ=0
760 PRINT"I NEED THE INITIALS OF THE";YY
770 PRINT"PLAYERS";:FOR I=1 TO YY
780 INPUT IN$(I):NEXT
790 IF YY=1 THEN P$(1)=IN$(1):GOTO 900
800 IF YY=2 THEN 1200
810 REM CHOOSE PLAYERS
820 REM 3 OR 4 PLAYING
830 A=1
840 AA=RND(Y):IF AA=A THEN 840
850 J=1:FOR I=2 TO YY
860 IF I=AA THEN 880
870 A(J)=I:J=J+1
880 NEXT
890 A(J)=A:A(J+1)=AA:GOTO 1200
900 PRINT"WE WILL NOW BEGIN. REMEMBER"
910 PRINT IN$(YY);", YOU ARE PLAYING"
920 PRINT"AGAINST ME, YOUR COMPUTER."
930 GOSUB 10000:CLS
940 GOTO 1400
970 PRINT"THERE ARE THREE OF YOU PLAYING."
980 PRINT"I WILL NOW CHOOSE TWO PEOPLE TO"
```

```
990 PRINT"PLAY ONE ANOTHER. THE PERSON"
1000 PRINT"REMAINING WILL PLAY THE"
1010 PRINT"WINNER OF THE FIRST ROUND..."
1020 PRINT
1090 GOSUB 10000 PRINT
1100 PRINT P$(1)," YOU WILL PLAY AGAINST"
1110 PRINT P$(2);". AFTER THESE HAVE PLAYED"
1120 PRINT"YOU WILL PLAY THE WINNER ";
1130 PRINT P$(3);" "
1140 GOTO 1310
1200 REM FOUR PLAYERS
1210 FOR I=1 TO YY
1215 IF YY=2 THEN P$(I)=IN$(I):GOTO 1230
1220 P$(I)=IN$(A(I))
1230 NEXT IF YY=2 THEN 1310
1235 IF YY=3 THEN 970
1240 PRINT P$(1);" YOU WILL PLAY ";P$(2)
1250 PRINT"IN THE FIRST ROUND. ";P$(3)
1260 PRINT"WILL PLAY ";P$(4);" IN THE"
1270 PRINT"SECOND  FINALLY, THE WINNER"
1280 PRINT"FROM EACH SET CAN PLAY EACH"
1290 PRINT"OTHER.":GOSUB 10000
1300 PRINT
1310 REM 2 PLAYERS
1320 A=RND(2)
1330 AA=A-1:IF AA=0 THEN AA=2
1335 IF A=1 THEN 1345
1340 P$(1)=P$(AA):P$(2)=P$(A):GOTO 1350
1345 P$(1)=P$(A):P$(2)=P$(AA)
1350 RR=A:WE=YY
1360 PRINT P$(RR);" YOU WILL PLAY FIRST."
1370 PRINT"YOU HAVE THE ";
1380 IF RR=2 PRINT"STEPPED PIECES":GOTO 1395
1390 PRINT"SOLID PIECES"
1395 GOSUB 10000:CLS
1400 REM PLAYING BOARD
1410 FOR X=10 TO 115
1420 SET(X,1):SET(X,2)
1430 SET(X,39):SET(X,40):NEXT
1440 FOR X=3 TO 38
1450 SET(10,X):SET(11,X)
1460 SET(114,X):SET(115,X):NEXT
1470 L=116:P=70:P1=P+128
1480 FOR I=1 TO 30
1490 PRINT@P,I;:P=P+10
1500 IF P>=L THEN P=P1:P1=P1+128:L=L+128
1510 NEXT:TP=5:BP=5:REM PIECES
1520 K=11:J=33:Y=3:L=Y+6
```

```
1530 FOR X=K TO J
1540 SET(X,Y):SET(X,Y+1):SET(X,Y+2)
1550 SET(X,Y+3):SET(X,Y+4):SET(X,Y+5)
1560 NEXT
1570 Y=Y+12
1580 IF Y>=39 THEN 1600
1590 GOTO 1530
1600 K=X:J=K+19:Y=L:L=Y-6
1610 IF Y=-3 THEN Y=-L
1620 IF Y=-9 THEN Y=-(L/3)-2
1630 IF Y=-15 THEN 1650
1640 GOTO 1530
1650 REM SET PIECES (TOP)
1660 X=35:Y=7:J=2:M=4
1670 FOR I=X TO X+16
1680 SET(I,Y):SET(I+40,Y)
1690 IF T=1 THEN SET(I+80,Y):M=10
1700 NEXT:IF T=1 THEN GOSUB 1900:GOTO 1720
1710 GOSUB 1900:X=15:Y=13:T=1:GOTO 1670
1720 REM SET PIECES (BOTTOM)
1730 GOSUB 1960:X=35:Y=Y+18:T=0:M=24
1740 FOR I=X TO X+16 STEP 2
1750 SET(I,Y):SET(I+40,Y)
1760 IF T=1 THEN SET(I+80,Y):M=30
1770 NEXT:IF T=1 THEN GOSUB 1900:GOTO 1790
1780 GOSUB 1900::X=15:Y=Y+6:T=1:GOTO 1740
1790 REM SET START LOCATIONS
1800 GOSUB 2030:J=2:FOR I=2 TO 10 STEP 2
1810 W(J)=1:W(J+10)=-99:J=J+2:NEXT:J=22
1820 FOR I=22 TO 30 STEP 2
1830 W(J)=99:J=J+2:NEXT:GOTO 2100
1840 GOSUB 10060:GOSUB 10080:PRINT@901,;
1850 INPUT"FROM SQUARE";F:IF F>30 THEN 1840
1855 IF F=0 AND RR=2 THEN RR=1:GOTO 4100
1856 IF F=0 AND RR=1 THEN RR=2:GOTO 4100
1860 GOSUB 10060:GOSUB 10080:PRINT@900,;
1870 INPUT"  TO SQUARE";T:IF T>30 THEN 1860
1880 RETURN
1900 REM PIECE LOCATION
1910 X(J)=X:Y(J)=Y
1920 J=J+2
1930 IF J=M THEN 1910
1940 IF J<=M AND T=1 THEN 1910
1950 RETURN
1960 REM PIECE LOCATION (MIDDLE)
1970 L=6:D=20:M=14
1980 X(J)=X+D:Y(J)=Y+L
1990 J=J+2
```

```
2000 IF J=M THEN 1980 ELSE T=1:M=20
2010 IF J<=M THEN D=0:L=12:GOTO 1980
2020 RETURN
2030 REM LOCATION PLUS AMOUNTS
2040 FOR I=8 TO 28 STEP 10:X(I)=X(I)+40
2050 NEXT:FOR I=4 TO 24 STEP 10
2060 X(I)=X(I)+40:NEXT
2070 FOR I=10 TO 30 STEP 10
2080 X(I)=X(I)+80:NEXT
2090 RETURN
2100 IF RR=3 THEN 2900
2104 GOSUB 1840:IF RR=2 THEN 2400
2105 IF W(F)=99 AND RR=1 THEN 4500
2108 Q=99
2110 IF W(T)<>-99 THEN 12000
2120 IF ABS(F-T)=6 THEN MX=1:GOTO 2145
2130 IF ABS(F-T)=12 THEN MX=2:A=6:GOTO 2145
2140 GOTO 2300
2145 IF W(T)=1 THEN 12000
2150 IF F<T THEN 2165
2160 A=-A
2165 IF MX=1 THEN 2190
2170 IF MX=2 AND W(F+A)=-99 THEN 12000
2180 IF MX=4 AND W(F+A)=-99 THEN 12000
2190 IF RR=2 OR RR=3 THEN J=2 ELSE J=1
2200 FOR I=X(F) TO X(F)+16 STEP J
2210 RESET(I,Y(F)):NEXT
2220 FOR I=X(T) TO X(T)+16 STEP J
2230 SET(I,Y(T)):NEXT
2235 IF RR=2 OR RR=3 THEN 2500
2240 IF MX=2 OR MX=4 THEN 2270
2250 W(T)=W(F):W(F)=-99
2260 GOTO 2280
2270 W(T)=W(F):W(F)=-99:W(F+A)=W(F):GOTO
     2700
2275 F=T:MX=0:KL=0:GOSUB 1860:ZC=1:GOTO 3120
2280 REM ALTERNATE PLAYER
2290 F=0:T=0:MX=0:A=0:Q=1
2295 IF WE>=2 THEN RR=2:GOTO 2100 ELSE RR=3:
     GOTO 2100
2300 IF ABS(F-T)=4 THEN MX=3:GOTO 2145
2310 IF ABS(F-T)=8 THEN MX=4:A=4:GOTO 2145
2320 GOTO 12000
2400 REM OPPONENT (STEPPED SQUARES)
2405 IF W(F)=1 AND RR=2 THEN 4500
2410 IF W(T)<>-99 THEN 12000
2420 IF ABS(F-T)=6 THEN MX=5:GOTO 2450
2430 IF ABS(F-T)=4 THEN MX=6:GOTO 2450
```

```
2440 GOTO 2580
2450 IF W(T)=99 THEN 12000
2460 IF F>T THEN 2190
2470 REM RESET SQUARE
2480 GOTO 2190
2500 REM INTERCHANGE SQUARES (STEPPED)
2510 W(T)=W(F):W(F)=-99
2515 IF KK=1 THEN 2540
2520 IF MX=7 OR MX=8 THEN 2540
2530 KK=0:MX=0:A=0:RR=1:MM=0:GOTO 2100
2540 IF F<T THEN A=-A:IF KK=1 THEN 2560
2550 W(F+A)=-99
2560 GOTO 2700
2565 IF KK=1 THEN F=T:IF RR=2 THEN GOSUB
     1860:ZC=1:GOTO 3120 ELSE I=F:GOTO 2925
2570 GOTO 2530
2580 IF ABS(F-T)=8 THEN MX=7:GOTO 2610
2590 IF ABS(F-T)=12 THEN MX=8:GOTO 2610
2600 GOTO 2100
2610 IF W(T)=99 THEN 12000
2620 IF MX=7 THEN A=-4:GOTO 2635
2630 A=-6:GOTO 2190
2635 IF RR=3 THEN 2190
2640 IF F<>16 AND F<>18 AND W(F+A)=-99
     THEN 2660
2650 GOTO 2190
2660 IF F<>20 AND F<>22 AND W(F+A)=-99
     THEN 12000
2670 GOTO 2190
2700 REM RESET SQUARE (JUMP)
2710 FOR I=X(F+A) TO X(F+A)+16
2720 RESET(I,Y(F+A)):NEXT
2725 W(F)=-99:W(F+A)=W(F):GOTO 4000
2730 REM MORE JUMPS ?
2740 REM TEST FORWARD AREAS
2750 I=T:A=0
2760 IF RR=3 THEN MM=1
2770 GOTO 2935
2900 GOSUB 10060:REM COMPUTERS PLAY
2905 MG=0:MX=0:EE=0:Q=1:GOSUB 10080
2910 FOR I=2 TO 30 STEP 2
2920 IF W(I)=99 THEN 2935
2930 NEXT:GOTO 3800
2935 IF I>=24 THEN 2950
2940 IF W(I+6)=Q AND W(I+12)=QQ THEN
     A=-12:GOTO 3100
2950 IF W(I+4)=Q AND W(I+8)=QQ THEN
     A=-8:GOTO 3100
```

```
2955 IF I<=10 THEN 2975
2960 IF W(I-4)=Q AND W(I-8)=QQ THEN
     A=8:GOTO 3100
2970 IF W(I-6)=Q AND W(I-12)=QQ THEN
     A=12:GOTO 3100
2975 IF RR=1 THEN 2290
2980 IF RR=2 THEN 2530
2990 IF MM=1 THEN MM=0:GOTO 2530
3000 A=0:GOTO 3700
3100 REM MOVE FEASABLE
3105 IF RR=1 THEN KL=1:GOTO 3260
3110 IF RR=2 THEN KK=1:GOTO 3260
3111 IF MM=1 THEN KK=1
3115 F=I:T=I-A:TT=T-A:IF TT<=0 TT=0
3120 IF F=2 AND T=10 THEN 3400
3130 IF F=4 AND T=16 THEN 3400
3135 IF ZC=1 THEN 3150
3140 IF F=6 THEN 3250
3150 IF F=10 AND T=16 THEN 3400
3152 IF ZC=1 THEN 3160
3155 IF F=10 THEN 3260
3160 IF F=12 AND T=20 THEN 3400
3162 IF F=14 AND T=26 THEN 3400
3165 IF F=14 AND T=6 THEN II=1:GOTO 3400
3166 IF ZC=1 THEN 3185
3170 IF F=14 THEN 3270
3180 IF F=16 THEN 3310
3185 IF F=16 AND T=22 THEN 3400
3186 IF F=16 AND T=22 THEN 3400
3188 IF ZC=1 THEN 3200
3190 IF F=20 THEN 3280
3200 IF F=22 AND T=10 THEN 3400
3201 IF F=22 AND T=16 THEN 3400
3202 IF F=22 AND T=30 THEN 3400
3205 IF F=26 AND T=14 THEN 3400
3210 IF F=24 AND T=16 THEN 3400
3211 IF F=30 AND T=22 THEN 3400
3215 IF F=30 AND T=16 THEN 3400 ELSE IF ZC=1
     THEN 3500
3220 IF F=26 THEN 3290
3230 IF F=30 THEN 3300
3240 IF F=22 AND T=30 THEN 3400 ELSE 3500
3250 IF T=10 OR T=14 THEN 3400 ELSE 3500
3260 IF T=2 OR T=6 OR T=22 OR T=24 THEN 3400
     ELSE 3500
3270 IF T=6 OR T=26 THEN II=1:GOTO 3400
     ELSE 3500
3280 IF T=12 OR T=16 OR T=26 THEN 3400
     ELSE 3500
```

```
3290 IF T=14 OR T=20 THEN 3400 ELSE 3500
3300 IF T=22 OR T=26 THEN 3400 ELSE 3500
3310 IF T=4 OR T=10 OR T=24 THEN 3400
3320 GOTO 3500
3400 REM WHICH TO WHERE
3410 IF RR=1 THEN 2290
3420 IF RR=2 THEN 2530
3430 IF II=1 THEN II=0:GOTO 2530
3440 GOTO 3830
3500 REM CONT. JUMP
3510 IF RR=1 AND ZC=0 THEN 2275
3515 IF RR=1 AND ZC=1 THEN ZC=0:GOTO 2108
3520 IF RR=2 AND ZC=0 THEN 2565
3525 IF RR=2 AND ZC=1 THEN ZC=0:GOTO 2420
3530 IF RR=3 THEN 3700
3700 REM CHECK AHEAD / RE-TEST (COMPUTER)
3710 IF W(F-A)<>-99 AND KK=0 THEN 2930
3715 IF A=8 OR A=-8 THEN MG=1
3720 IF A=12 OR A=-12 THEN MG=1
3730 IF MG=1 THEN MG=0:KK=0:GOTO 2410
3740 IF I>=8 AND KK=0 AND EE=0 THEN 3780
3750 MX=MX+1
3760 IF MX<=1 THEN 2910
3770 GOTO 2410
3780 IF BP>=4 AND W(TT)=1 THEN 2925
3790 GOTO 3750
3800 REM ALTERNATE MOVES
3810 AD=1:MM=2:FOR I=30 TO 2 STEP-2
3820 IF W(I)=99 THEN EE=1:GOTO 3855
3830 IF KK=1 THEN 2410 ELSE NEXT
3840 RR=0:GOTO 4110
3850 KK=0:F=0:T=0:RR=1:A=0:GOTO 2100
3855 IF I<=4 THEN 3875
3860 IF W(I-4)=QQ THEN A=4:GOTO 3100
3870 IF W(I-6)=QQ THEN A=6:GOTO 3100
3875 IF I>=26 THEN 3900
3880 IF W(I+4)=QQ THEN A=-4:GOTO 3100
3890 IF W(I+6)=QQ THEN A=-6:GOTO 3100
3900 GOTO 3830
4000 REM GAME COMPLETED ?
4010 IF RR=1 THEN 4040
4020 IF RR=2 OR RR=3 THEN 4070
4030 GOTO 2730
4040 BP=BP-1
4050 IF BP<=0 THEN 4100
4060 GOTO 2730
4070 TP=TP-1
4080 IF TP<=0 THEN 4100
```

```
4090 GOTO 2730
4100 REM COMPLETE
4110 GOSUB 10030
4120 IF RR=2 THEN 4190
4130 IF RR=3 THEN 4190
4140 IF GL=1 THEN 4170
4150 PRINT"YOU LOSE.....";P$(2)
4160 P$(FF)=P$(1):GOTO 4210
4170 PRINT"I HAVE LOST...":REM COMPUTER
4180 GOTO 4210
4190 PRINT"YOU HAVE LOST...";P$(1)
4200 P$(FF)=P$(2)
4210 REM CONTINUE / END
4220 IF WE<=2 THEN 4420
4230 REM 3 OR 4 PLAYERS
4240 WN$(FF)=P$(FF):FF=FF+1
4245 IF AQ=1 THEN 4410
4250 IF FF=3 THEN 4400
4260 IF WE=3 THEN P$(2)=P$(3):GOTO 4300
4270 P$(1)=P$(3):P$(2)=P$(4)
4280 PRINT P$(1);" YOU WILL NOW PLAY ";P$(2)
4290 GOTO 1310
4300 PRINT P$(FF);" YOU WILL NOW PLAY ";
4310 PRINT P$(1)
4320 GOTO 1310
4400 REM FINAL
4405 IF WE=4 THEN 4450
4410 PRINT WN$(FF-1);" IS THE GRAND WINNER !!"
4420 PRINT
4430 PRINT"END OF PROGRAM RUN..."
4440 END
4450 AQ=1:P$(1)=WN$(1):P$(2)=WN$(2)
4460 PRINT P$(1);" YOU WILL PLAY ";P$(2)
4470 PRINT"TO DETERMINE THE FINAL WINNER."
4480 GOTO 1310
4500 REM NOT CORRECT PLAYER
4510 GOSUB 10030
4520 PRINT"IT IS NOT YOUR PLAY !!"
4530 GOSUB 10050
4540 GOTO 2100
10000 REM TIME LOOP
10010 FOR Q=1 TO 5000:NEXT
10020 RETURN
10030 PRINT@901,;
10040 RETURN
10050 FOR U=1 TO 1000:NEXT
10060 PRINT@901,STRING$(50,32)
10070 IF T=0 RETURN
```

```
10080 PRINT@965,"PLAYER . . .";P$(RR);
10085 IF RR=3 PRINT"COMPUTER";
10090 RETURN
11000 GOSUB 10030
11005 IF ZC=1 THEN 11030
11010 PRINT"** CANNOT RESPOND **"
11020 GOSUB 10050
11030 F=0:T=0:RETURN
12000 REM ERROR MESSAGE
12010 IF RR=3 THEN 2905
12020 GOSUB 11000
12030 GOTO 2104
```

Variables and Strings

X,Y & W - Subscripted for piece set and reset locations, and the contents of squares W(I)
QQ - Used to test ahead to determine the status of a square.
FF - Used for player, if more than one is playing
YY - Number of players
TN - Used to skip the instructions
AA & A - Used for random selection of player's turn
RR - Used for player: RR=1 is for solid pieces, and RR=2 is for stepped playing pieces
IN$(I) - Player's initials
P$(I) - Player's initials, after random selection of first player
WZ - Total number of players
L,P & P1 - Used for setting up the game board, along with X,Y,J & K
F & T - For inputs of numbers of the squares to move from and to
MX - Status of playing piece, to test if legal
J - Used again for stepped pieces if RR=2 or RR=3
KL & ZC - Used to determine if a jump is legal
KK & ZC - Same as above for stepped pieces
A - For movement of piece forward or backward
EE, MG & AD - For alternate move (if possible) by computer
TP - Number of squares at the top of the field
BP - Number of squares at the bottom of the field
GL - Computer playing
WN$(FF) - Used if there are 3 or 4 players to determine the winner
Q & U - For time loop (pause)

Explanation of Program Lines

Line 40 dimensions the X,Y, and W arrays for proper square and piece locations.

Line 50 sets the variable QQ to −99 for piece identification for a possible jump.

Line 60 sets the variables FF to 1. FF is used to determine the final winner when there are more than 2 players.

Lines 70-150 ask if instructions are needed. If they are not TN will equal 1 and the instructions will be skipped.

Lines 160-190 ask for the number of player's amount. If there is only 1, GL will equal 1, and the computer will play against the user.

Lines 200-210 determine whether or not the instructions are needed. If they are not, the program branches to line 510.

Lines 220-510 display the instructions for play

Line 520 tests the value of YY (number of players). If YY is equal to 1, program branches to line 540.

Line 530 sends program control to the appropriate lines for 2 or more players.

Lines 540-600 determine who will play first, the user or the computer.

Lines 610-780 let user enter each player's initials.

Lines 790-800 send program control to the appropriate lines if only 1 or 2 are playing.

Lines 810-890 randomly select players (3 or 4) to determine who will play against whom.

Lines 900-940 display a message for one player.

Lines 970-1140 display a message indicating who will play against whom when 3 people are playing.

Lines 1200-1300 display a message indicating who will play against whom when 4 people are playing.

Lines 1310-1395 select who will play first and which type of playing piece he or she will have.

Lines 1400-1830 set the graphics for the playing field (board) and all of the pieces. They determine the number of pieces and the value of each place: is assigned to the top pieces, 99 to the bottom pieces, and −99 to blanks.

Lines 1840-1880, accept player inputs in response to the FROM and TO prompts. If FROM (F) is equal to 0 (zero), the player has just thrown in the towel, and the program will branch to either the termination routine or next player routine.

Lines 1900-2090 establish the pieces and their locations on the playing board.

Lines 2100-2270 along with lines 2300-2320 are used to move the solid pieces.

Lines 2190-2240 are also used to move the stepped pieces.

Lines 2250-2270 are used to redefine the contents of the squares that were involved in the FROM and TO movement.

Line 2275 are used for a possible double (or more) jump.

Lines 2280-2295 allow the next player or the computer to input moves if RR = 3.

Lines 2400-2660 work much the same as lines 2100-2270 above, except these are for the stepped pieces.

Lines 2700-2725 reset any square that has been jumped and delete one piece from the player that was jumped. Lines 4000-4090 are used to determine whether or not the game is over.

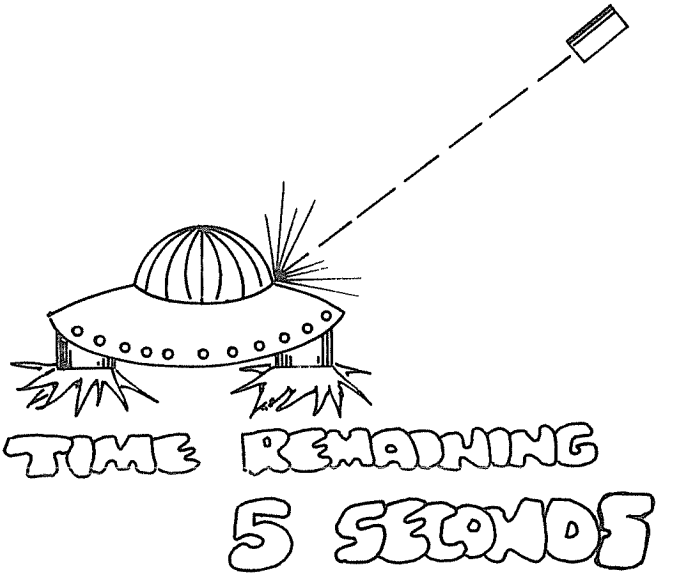
Lines 2730-2770 are used to determine if another jump might be possible. For this the program branches to line 2935-3530. If any of the conditions are met, the user will be allowed another jump.

Lines 2900-3530 are for the computer's pieces as well. These lines will determine if the computer can jump or just move. If no jumps can be made, lines 3700-3900 are used for movement from one square to another. If none of the conditions are met following the for-next loop beginning at line 3810 and ending at line 3830, the computer will terminate the program at line 3340.

Lines 4100-4480 determine who won and who will play against whom in the next game. If all conditions are met, the grand winner is printed and the program terminated.

Lines 4500-4540 prints message if a player is trying to move someone's pieces other than his or her own.

Lines 10000-12030 contain the necessary time loops, the turn determination routine, message CANNOT RESPOND if an illegal move is entered.

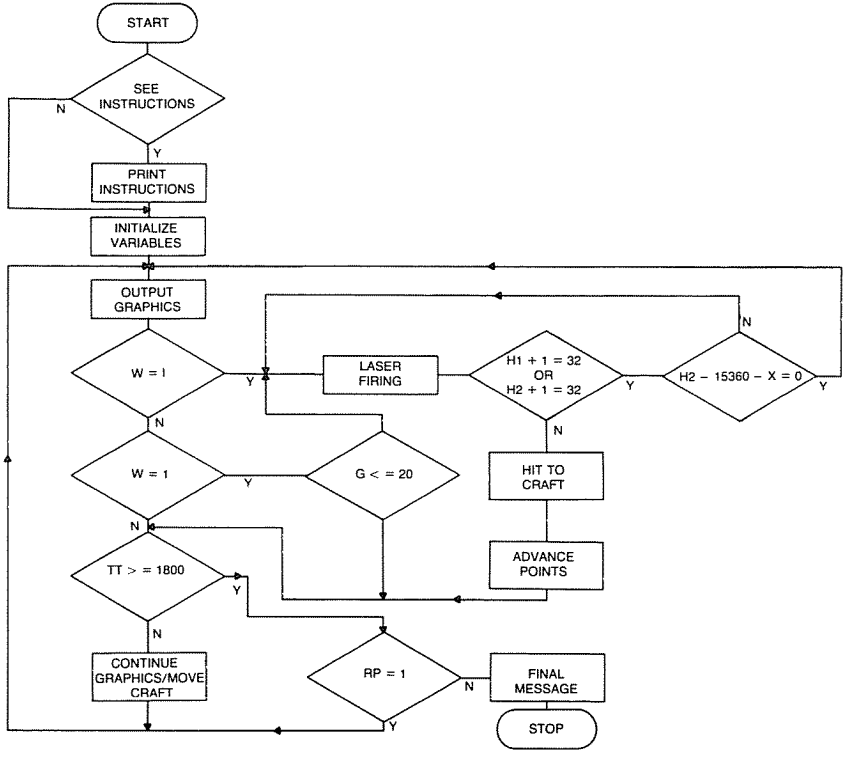


HOLD TIME

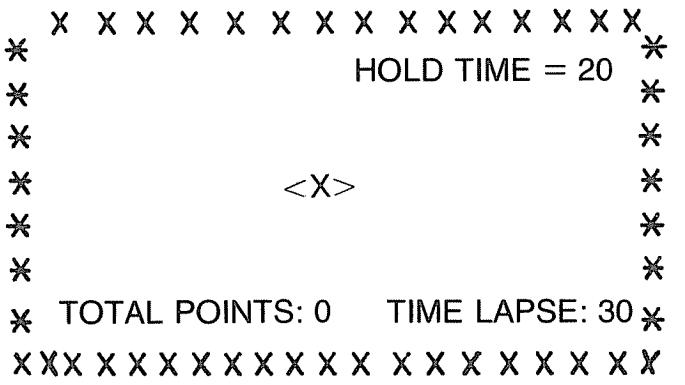
This game enables you to make all computer movement stop while you try and destroy the computer's piece. All your decisions will have to be exact if you plan on doing this. To destroy the computer's piece, your lasers must be at the right place at the right time. The amount of time you can hold the computer's piece in one location is 20 seconds. A long time you say? Run the program and see for yourself. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B. Remember to use the proper value for SS in line 295. Refer to page 3.

Sample Run

```
HOLD TIME...

INSTRUCTIONS? YES

HOLD TIME...A MATTER OF HIGH OR
LOW POINT TOTALS. THE COMPUTER'S
CRAFT WILL BLIP ALL AROUND YOUR
VIDEO, DOWN THE LEFT AND RIGHT
SIDES WILL BE ASTERISKS. THESE
ARE YOUR LASERS. AT THE INSTANT
```



Flowchart for *Hold Time*.



YOU THINK THE CRAFT IS IN LINE
WITH THESE, HIT THE 'CLEAR' KEY.
YOUR LASERS WILL START MOVING
TOWARD THE CENTER, THE COMPUTER
CRAFT WILL ALSO CONTINUE TO MOVE

PRESS ENTER?

YOU CAN 'HOLD' THE COMPUTERS'
CRAFT FROM MOVING FOR AN AMOUNT
OF TIME, ROUGHLY 20 SECONDS.
YOUR SCORE WILL RANGE FROM 1500 TO 400, DEPENDING ON WHERE YOU
DESTROY THE CRAFT (TOP OR BOTTOM
OF THE VIDEO). THE TIME LIMIT FOR THE
ENTIRE RUN WILL BE JUST ABOUT
FIVE MINUTES. THIS WILL BE NOTED
AT THE LOWER RIGHT OF YOUR VIDEO.
THE COUNT STARTS AT 30. OH YES, TO 'HOLD' THE COMPUTERS'
CRAFT, HOLD DOWN THE 'ENTER'
KEY.....

PRESS ENTER TO BEGIN?

Program Listing

```
100 REM PROGRAM TITLE: HOLD TIME
110 CLS
```

```
115 CLEAR 350
120 RANDOM
125 PRINT"HOLD TIME..."
130 INPUT"INSTRUCTIONS (Y/N)";Z$
135 IF Z$="N" THEN CLS:GOTO 290
140 PRINT
145 PRINT"HOLD TIME...A MATTER OF HIGH OR"
150 PRINT"LOW POINT TOTALS. THE COMPUTER'S"
155 PRINT"CRAFT WILL BLIP ALL AROUND YOUR"
160 PRINT"VIDEO. DOWN THE LEFT AND RIGHT"
165 PRINT"SIDES WILL BE ASTERISKS. THESE"
170 PRINT"ARE YOUR LASERS. AT THE INSTANT"
175 PRINT"YOU THINK THE CRAFT IS IN LINE"
180 PRINT"WITH THESE, HIT THE 'CLEAR' KEY."
185 PRINT"YOUR LASERS WILL START MOVING"
190 PRINT"TOWARD THE CENTER. THE COMPUTER"
195 PRINT"CRAFT WILL ALSO CONTINUE TO MOVE."
200 INPUT"PRESS ENTER";X$:CLS:PRINT
205 PRINT"YOU CAN HOLD THE COMPUTER'S"
210 PRINT"CRAFT FROM MOVING FOR AN AMOUNT"
215 PRINT"OF TIME, ROUGHLY 20 SECONDS."
220 PRINT"YOUR SCORE WILL RANGE FROM 1500"
225 PRINT"TO 400, DEPENDING ON WHERE YOU"
230 PRINT"DESTROY THE CRAFT (TOP OR BOTTOM"
235 PRINT"OF THE VIDEO). THE TIME LIMIT FOR
    THE"
240 PRINT"ENTIRE RUN WILL BE JUST ABOUT"
245 PRINT"FIVE MINUTES. THIS WILL BE NOTED"
250 PRINT"AT THE LOWER RIGHT OF YOUR VIDEO."
255 PRINT"THE COUNT STARTS AT 30."
260 PRINT"OH YES, TO HOLD THE COMPUTER'S"
265 PRINT"CRAFT, HOLD DOWN THE 'ENTER'"
270 PRINT"KEY....."
275 INPUT"PRESS ENTER TO BEGIN";X$:CLS
290 REM CRAFT/SOUND LOCATION
295 C$="(X)":SS=31266
300 REM OUTER GRAPHICS/POINTS
305 Q$=STRING$(63,42):M=474
310 G=0:PI=16:PRINT@0,Q$;:PRINT@960,Q$;
315 PRINT@121,"20";:FOR X=64 TO 896 STEP 64
320 PRINT@X,LEFT$(Q$,1);
325 PRINT@X+63,LEFT$(Q$,1);:GOTO 460
330 IF MM<>0 GOSUB 355:NEXT ELSE NEXT
335 PRINT@0,CHR$(30);:PRINT@960,CHR$(30);
340 FOR X=64 TO 896 STEP 64
345 PRINT@X,CHR$(32);:PRINT@X+63,CHR$(32);
350 GOSUB 355:NEXT:GOSUB 355:GOTO310
355 PRINT@954,ABS(INT(TT/60)-30);
```

```
356 PRINT@MM,STRING$(5,32);
360 PRINT@M,C$;:TT=TT+1
365 IF TT>=1800 THEN 800
370 MM=M:W=PEEK(14400)
375 IF W=1 AND G<=20 THEN G=G+.4:GOSUB
    750:RETURN
380 XX=RND(4):REM CRAFT MOVE
385 ON XX GOTO 390,400,410,420
390 REM LEFT MOVE
395 M=M-RND(5):GOTO 445
400 REM RIGHT MOVE
405 M=M+RND(5):GOTO 445
410 REM UP MOVE
415 M=M-64:GOTO 445
420 REM DOWN MOVE
425 M=M+64:IF PEEK(16258)=32 GOSUB 670
430 IF PEEK(16303)=32 GOSUB 670
435 IF PEEK(15426)=32 GOSUB 670
440 IF PEEK(15470)=32 GOSUB 670
445 IF M<=66 THEN 420
450 IF M>=950 THEN 410
455 POKE SS,13:NN=USR(0):RETURN
460 REM FIRE
465 IF X<=832 THEN W=PEEK(14400)
470 PI=PI-1:IF W=0 THEN 330
475 IF W=2 THEN 485
480 GOTO 330
485 REM LASER ADVANCE
490 IF X>=896 OR X<=127 THEN 330
495 H=X:H1=H+15360:H2=H1+63:UN=UN+1:S2=100
500 IF RP=1 PRINT@81,ABS(R-UN);
505 POKE H1,32:POKE H2,32:POKE SS,S2
510 IF PEEK(H1+1)<>32 OR PEEK(H2-1)<>32 THEN 535
515 H1=H1+1:H2=H2-1:S2=S2-1:NN=USR(0)
520 POKE H1,42:POKE H2,42
525 IF ABS(H2-15360-X)=0 THEN 850
530 GOSUB 355:GOTO 505
535 REM DIRECT HIT
540 POKE H1,42:POKE SS,14
545 IF PEEK(H2-1)<>32 THEN 565
550 POKE H2,32:H2=H2-1
555 POKE H2,42
560 NN=USR(0):GOTO 545
565 REM DELETE
570 Z=1:POKE H1,32:POKE H2,32
575 N=M+15360:S1-220
580 POKE N,32:POKE N+1,32
585 POKE N+2,32
```

```
590 J=RND(5)
595 IF N+J+64<=16319 THEN 605
600 GOTO 625
605 POKE N+J+64,42:POKE N+J-64-J,42
610 FOR I=1 TO J*10:NEXT
615 POKE N+J+64,32:POKE N+J-64-J,32
620 GOTO 640
625 POKE N+J-64,42:POKE N+J-64-J,42
630 FOR I= TO J*10:NEXT
635 POKE N+J-64,32:POKE N+J-64-J,32
640 IF Z<=7 THEN 655
645 POKE SS,255:NN=USR(0)
650 IF RP=0 GOTO 660 ELSE PI=PI+1:GOTO 665
655 Z=Z+1:POKE SS,S1:NN=USR(0):GOTO 590
660 PI=PI+1:X$=X$+"+"
665 PP=PP+PI*100:GOSUB 670:GOTO 850
670 A$="TOTAL POINTS:":T$="TIME LAPSE:"
675 F=898
680 FOR I=1 TO LEN(A$)
685 PRINT@F+45,MID$(T$,I,1);
690 PRINT@F,MID$(A$,I,1);
695 F=F+1:NEXT
700 PRINT PP;:IF RP=1 THEN 710
705 IF X$=""RETURN
710 REM RESERVE POWER
715 R$="RESERVE POWER:"
720 F=66
725 FOR I=1 TO LEN(R$)
730 PRINT@F,MID$(R$,I,1);
735 F=F+1:NEXT
740 IF RP=1 THEN X$=STR$(ABS(R-UN))
745 PRINT X$;
750 TI$="HOLD TIME:":REM HOLD TIME
755 IF PEEK(15470)<>32 THEN 770
760 F=110:FOR I=1 TO LEN(TI$)
765 PRINT@F,MID$(TI$,I,1);:F=F+1:NEXT
770 PRINT@120,INT(ABS(G-20));
775 RETURN
780 REM TIME REMAINING
785 IF RP=1 THEN DK=1:GOTO 860
790 IF TT>=1800 THEN 800
795 GOTO 300
800 FOR TI=1 TO 1200:NEXT
805 CLS:R=LEN(X$)
810 IF R=0 THEN 910
815 PRINT"VERY WELL DONE...."
820 PRINT"YOU HAVE ";R;"UNITS OF"
825 PRINT"RESERVE POWER...."
```

```
830 PRINT"PRESS ENTER TO USE";
835 INPUT X$:TT=0:RP=1:UN=0
840 CLS
845 GOTO 300
850 REM UNITS DELETE (RESERVE)
855 IF RP=0 THEN 330
860 IF UN>=R THEN 875
865 IF DK=0 THEN 330
870 DK=0:GOTO 790

875 REM FINISH
880 FOR TI=1 TO 1200:NEXT
885 CLS
890 PRINT"RESERVE POWER EXHAUSTED...."
895 PRINT"TOTAL POINTS ACCUMULATED:";PP
900 PRINT
905 GOTO 930
910 PRINT"YOU HAVE DESTROYED NOTHING."
915 PRINT"YOU HAVE NO RESERVE POWER."
920 PRINT
925 GOTO 895
930 PRINT
935 PRINT"END OF RUN."
990 END
```

Variables and Strings

C$ - Computer's craft
SS - Sound location (machine routine)
Q$ - Outer graphics, asterisks
M,MM - Location of computer craft
PI - Points, total times 100
G - Hold time—holding of computer craft
TT - Amount of playing time
W - Key closure (PEEK)
XX - Random craft movement
NN - USR Call for sound output
H - Location of asterisk when laser fired
H1,H2 - Laser advanced toward center then outward
S2 - Also for sound output
UN - Reserve units used, not valid until RP=1
RP - On reserve power
Z - General, for flashing of destroyed craft
N - Location of computer craft when destroyed
J - A random amount used to flash asterisks above or below craft
X$ - Reserve units, if user's craft destroyed

PP - Total outstanding points
F - Print @ locations, messages
A$,T$,R$, & TI$ - Messages
R - Reserve units, total
DK - Finish of game

Explanation of the Program Lines

Line 115 clears 350 bytes of memory for string storage.

Lines 125-275 display the instructions for playing.

Line 295 sets C$ to the computer's craft and initializes the variable SS for the sound routine.

Lines 300-430 print the graphics near the edges of the screen and control the movement of the computer's craft. The craft is moved a random amount distance, determined between lines 370 and 415. Line 366 tests whether the enter key is being held and variable G is less than or equal to 20. Variable TT (line 362 is checked to see if the time has expired for the run. The peek statements in lines 415-418 are used for the print messages. If the computer's craft happens to blank out any letters of these messages, they will be reprinted as soon as possible. Lines 420 and 425 keep the computer's craft within limits.

Line 430 outputs a beep sound each time the craft moves.

Lines 500-565 enable the user to fire the laser. Here the asterisks are moved toward the center of the video. If a hit is detected (line 545) the program branches to line 570. Line 565 continues the movement of the computer's craft, if the Enter key is not being held down. Line 560 tests the value of H2 for stopping the laser attack.

Lines 570-676 and 685 detect or destroy the craft and flashes asterisks at its location.

Line 680 tests whether or not user is on reserve power.

Lines 690-695 increase the variable PI by 1 and then multiply that amount by 100 to determine your score (line 695). X$ in line 690 is then increased by one + (plus sign) to indicate added units of reserve power.

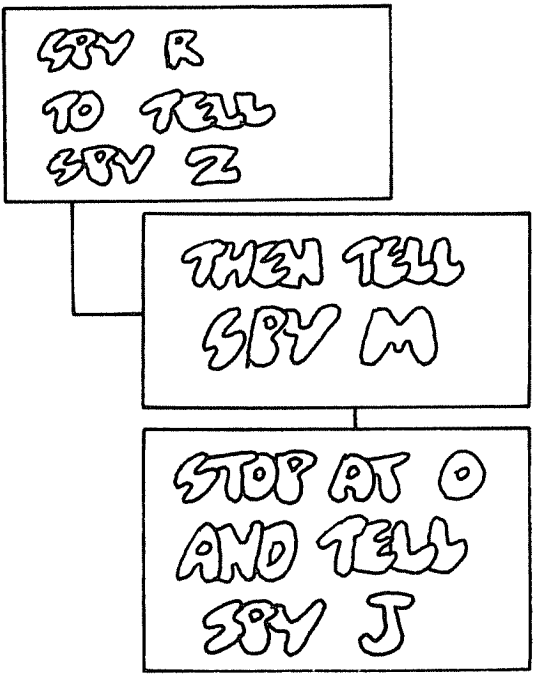
Lines 700-795 print messages and amounts.

Lines 800-830 again test the value TT to determine whether or not the game is over.

Lines 835-875 tells the user how many units of reserve power he or she has. (The player receives one unit for each computer craft destroyed). Then the user presses the enter key to begin using the reserve laser power.

Lines 900-920 test the amount of UN (reserve power) to see if it has all been used.

Lines 925-990 display the final message and the total score.



DANGEROUS AFFAIR

All of your life you've wanted to be a spy, a double agent that would help his country in time of need. Here's your chance. Through messages you will relay, the computer will test to see if you would actually make a good spy. Some of the messages will be straight while others will be coded. There are agents (besides you) that will accidently forget the message you've given them, so try to remember each...very, very well. Remember to use the proper value for L in line 250. Refer to page 3.

Sample Run

```
>> DANGEROUS AFFAIR <<

SEE OUTLINE? MAYBE
TRY A YES OR A NO ANSWER.

SEE OUTLINE? YES
SO YOU WANT TO BE A SPY AND YOU
THINK YOU'D BE A GOOD ONE AT
```

THAT? THIS PROGRAM WILL SOON
LET YOU KNOW. YOU MUST FOLLOW
AND PASS ALONG ANY INSTRUCTIONS THAT
WILL BE GIVEN TO YOU; SOME
OF THEM WILL BE CODED, WHICH
MEANS YOU'LL HAVE TO BREAK
THE CODE BEFORE PASSING THE
INFORMATION GIVEN.
REMEMBER TO PASS INFORMATION
EXACTLY AS IT IS GIVEN TO YOU
(EXCEPT WHERE CODED),
FIRST NAME? 005

PASS THIS ALONG:
.M.P 4 TA .TS NIAM OT OG
WHAT WAS THAT MESSAGE? 005
PLEASE REPEAT

PASS THIS ALONG:
NADES KCALB A NI TEG

WHAT WAS THAT MESSAGE? 005
I MISSED PART OF IT

PASS THIS ALONG:
ANSWER THE PHONE AT 4TH ST.

WHAT WAS THAT MESSAGE? 005

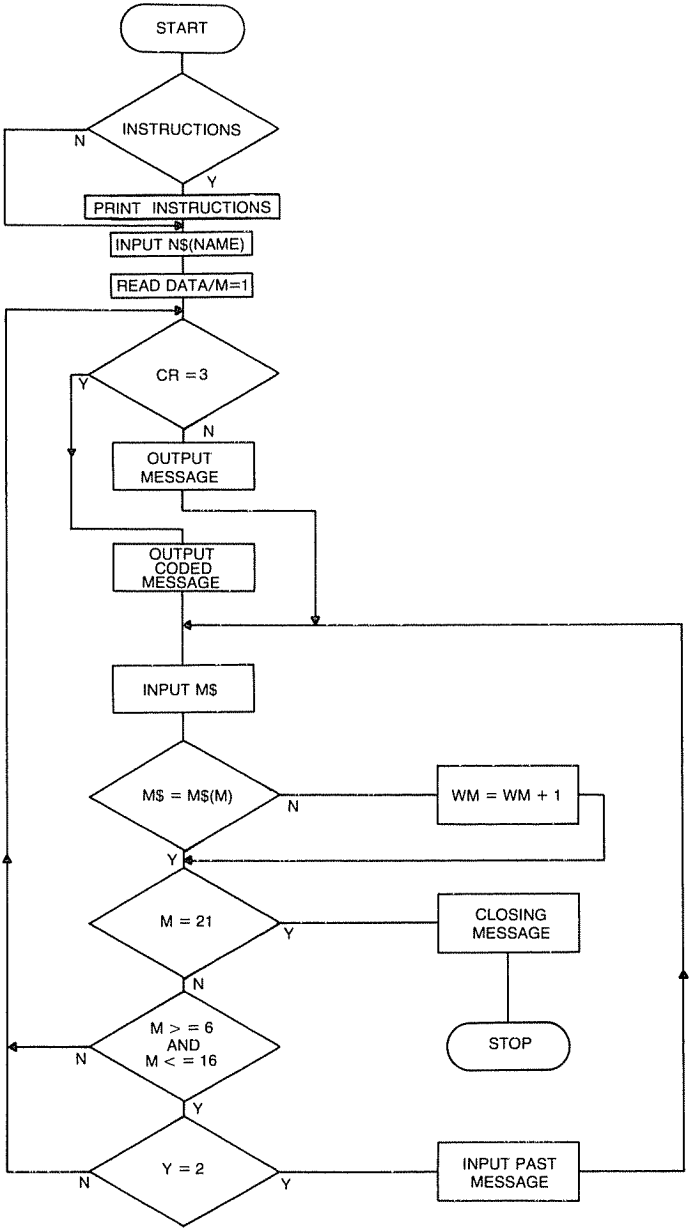
ANSWER A PHONE AT 4TH STREET

PASS THIS ALONG:
GO TO THE 12TH ST. BRIDGE

WHAT WAS THAT MESSAGE? 005
GO TO A BRIDGE AT 12TH STREET

PASS THIS ALONG:
NOITCES F/D ROF ELIF AERA TEG

WHAT WAS THAT MESSAGE? 005
SOMETHING ABOUT A NOTICE?



Flowchart for *Dangerous Affair*.

AGENT 28 DID NOT RECEIVE
MESSAGE # 5, WHAT WAS
IT, 005?
YOU TELL ME
PASS THIS ALONG:
MEET HARRY AT 1ST ST. BAR

WHAT WAS THAT MESSAGE? 005
MEET HARRY AT BAR

PASS THIS ALONG:
YRRAH FO ESOPSID

WHAT WAS THAT MESSAGE? 005
I FORGOT REPEAT O.K.?

STOP

This was not a complete run.

Program Listing

```
10 REM PROGRAM TITLE: DANGEROUS AFFAIR
20 RANDOM:CLS
30 CLEAR 500:DIM M$(21)
50 PRINT"›› DANGEROUS AFFAIR ‹‹"
60 PRINT
70 INPUT"SEE OUTLINE";X$
80 IF X$="YES" THEN 120
90 IF X$="NO" THEN PRINT:GOTO 240
100 PRINT"TRY A YES OR A NO ANSWER."
110 GOTO 60
120 PRINT"SO YOU WANT TO BE A SPY AND YOU"
130 PRINT"THINK YOU'D BE A GOOD ONE AT"
140 PRINT"THAT? THIS PROGRAM WILL SOON"
150 PRINT"LET YOU KNOW. YOU MUST FOLLOW"
160 PRINT"AND PASS ALONG ANY INSTRUCTIONS
    THAT"
170 PRINT"WILL BE GIVEN TO YOU; SOME"
180 PRINT"OF THEM WILL BE CODED, WHICH"
190 PRINT"MEANS YOU'LL HAVE TO BREAK"
200 PRINT"THE CODE BEFORE PASSING THE"
210 PRINT"INFORMATION GIVEN."
220 PRINT"REMEMBER TO PASS INFORMATION"
```

```
230 PRINT"EXACTLY AS IT IS GIVEN TO YOU"
235 PRINT"(EXCEPT WHERE CODED)."
240 INPUT"FIRST NAME";N$
250 CLS:M=1:L=31266
260 REM GET MESSAGES (INFORMATION)
270 FOR I=1 TO 20
280 READ M$(I):NEXT
290 DATA GO TO MAIN ST. AT 4 P.M.
300 DATA GET IN A BLACK SEDAN
310 DATA ANSWER THE PHONE AT 4TH ST.
320 DATA GO TO THE 12TH ST. BRIDGE
330 DATA GET AREA FILE FOR D/F SECTION
340 DATA MEET HARRY AT 1ST ST. BAR
350 DATA DISPOSE OF HARRY
360 DATA GIVE THE D/F FILE TO INTER C
370 DATA RETURN TO MAIN ST. AT 11 P.M.
380 DATA GET IN BLUE/WHITE HARDTOP
390 DATA RECEIVE CODED MESSAGE FROM DRIVER
400 DATA GIVE MESSAGE TO RHONDA
410 DATA DUMP HER THEN STEAL HER CAR
420 DATA DRIVE IT TO 31ST ST. EXIT
430 DATA DESTROY THE CAR
440 DATA GO TO LAST FLIGHT AIRPORT
450 DATA BOARD THE 3 A.M. FLIGHT
460 DATA LEAVE PLANE AT LYTERA AIRPORT
470 DATA GIVE CODED MESSAGE TO MAN AT GATE
480 DATA RETURN TO PLANE AT 3 P.M.
490 PRINT"PASS THIS ALONG:"
500 FOR Y=1 TO 2
510 CR=RND(3):IF CR=3 THEN 590
520 NEXT:I=1:J=LEN(M$(M))
530 PRINT MID$(M$(M),I,1);
540 POKE L,ASC(MID$(M$(M),I,1))+45
550 A=USR(0)
560 I=I+1
570 IF I>J THEN 700
580 GOTO 530
590 REM CODE IT
600 FOR Y=1 TO 2:CC=RND(2)
610 NEXT
620 I=LEN(M$(M)):J=1
630 IF CC=2 THEN 690
640 PRINT MID$(M$(M),I,1);
650 POKE L,ASC(MID$(M$(M),I,1))+60
660 A=USR(0)
670 I=I-1:IF I=0 THEN 700
680 GOTO 630
690 K$=STR$(ASC(MID$(M$(M),I,1)))
```

```
695 PRINT K$;:GOTO 650
700 REM PAUSE
710 IF CC=1 OR CC=2 THEN 800
720 FOR W=1 TO 400:NEXT:CLS
730 PRINT"WHAT WAS THAT MESSAGE ";N$
740 INPUT M$
750 IF M$<>M$(M) THEN WM=WM+1
760 M=M+1
770 IF M=21 THEN 1000
780 FOR W=1 TO 1000:NEXT
790 CLS:GOTO 850
800 REM LONGER PAUSE
810 FOR W=1 TO 2500:NEXT
820 CR=0:CC=0:GOTO 720
850 REM DID NOT RECEIVE OR FORGOTTEN
860 IF M>=6 AND M<=16 THEN 880
870 GOTO 490
880 Y=RND(2)
890 IF Y=1 THEN 490
900 Y=RND(M):AG=RND(100)
910 PRINT"AGENT";AG;"DID NOT RECEIVE"
920 PRINT"MESSAGE #";Y;", WHAT WAS"
930 PRINT"IT ";N$
940 INPUT M$
950 IF M$<>M$(Y) THEN WM=WM+1
960 CLS:GOTO 490
1000 REM END
1010 IF WM>=10 THEN 1070
1020 PRINT"YOU'VE DONE ALRIGHT ";N$
1030 PRINT"YOU NOW CAN BECOME A"
1040 PRINT"COMPUTER SPY, AND"
1050 PRINT"PASS MESSAGES..."
1060 GOTO 1130
1070 PRINT"GOOD GRIEF, TURKEY !!!"
1080 PRINT"YOU FAILED";WM;"OF THE";M-1
1090 PRINT"MESSAGES GIVEN TO YOU."
1100 PRINT"TO YOU THIS WAS NOT A"
1110 PRINT"DANGEROUS AFFAIR; IT WAS"
1120 PRINT"A LOST MISSION !!!"
1130 PRINT
1140 PRINT"END OF PROGRAM..."
1150 END
```

Variables and Strings

N$ - First name of user
M$(I) - Information messages
M - Counter, messages given

L - Machine language address
CR,CC - Used to code message
J - Length of M$(M), which is the message if the message is not coded.
If the message is coded, I is the Length of M$(M)
M$ - User input of message
WM - Wrong message entered

Explanation of the Program Lines

Line 30 clears 500 bytes of memory for string storage and dimensions M$.

Lines 50-235 display the instructions, if they are required.

Line 240 inputs the user's first name.

Line 250 clears the video and initializes the variables M and L.

Lines 260-280 read the data elements into the M$ array.

Line 490 begins the actual entries.

Lines 500-510 branch to the message encoding routine if CR is equal to 3.

Line 520 initializes the variables I and J.

Lines 530-580 display the message in uncoded form and produce a sound for each letter.

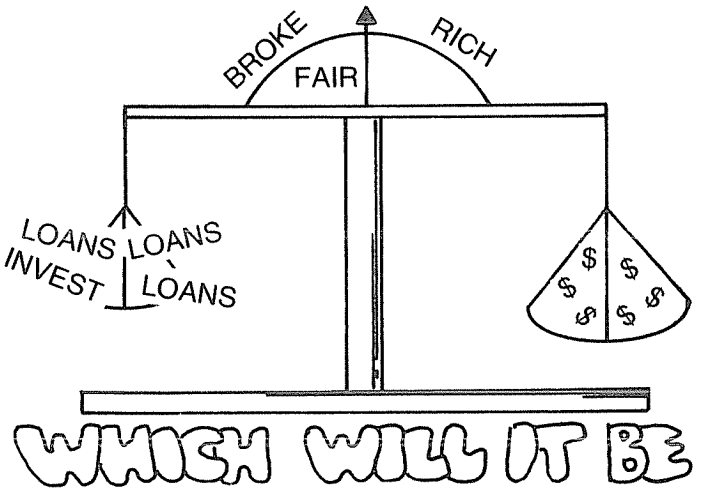
Lines 590-695 output the messages in simple coded form (if CR = 3). The message will be printed in one of two ways, depending on the value of CC.

Lines 700-720 allow a pause before the video is cleared. A longer pause is allowed if message was printed in code.

Lines 730-790 allow the user to pass the message he or she was just given. If the message is incorrect, the variable WM will be increased by 1. If all messages have been used (M = 21), the program branches to line 1000.

Lines 850-960 further test the user's memory by asking him to repeat earlier messages, if M>= 6, M<= 16, and Y (random is equal to 2. Again, if the response is incorrect, counter WM will be increased by 1.

Lines 1000-1150 print a final message and terminate the program.

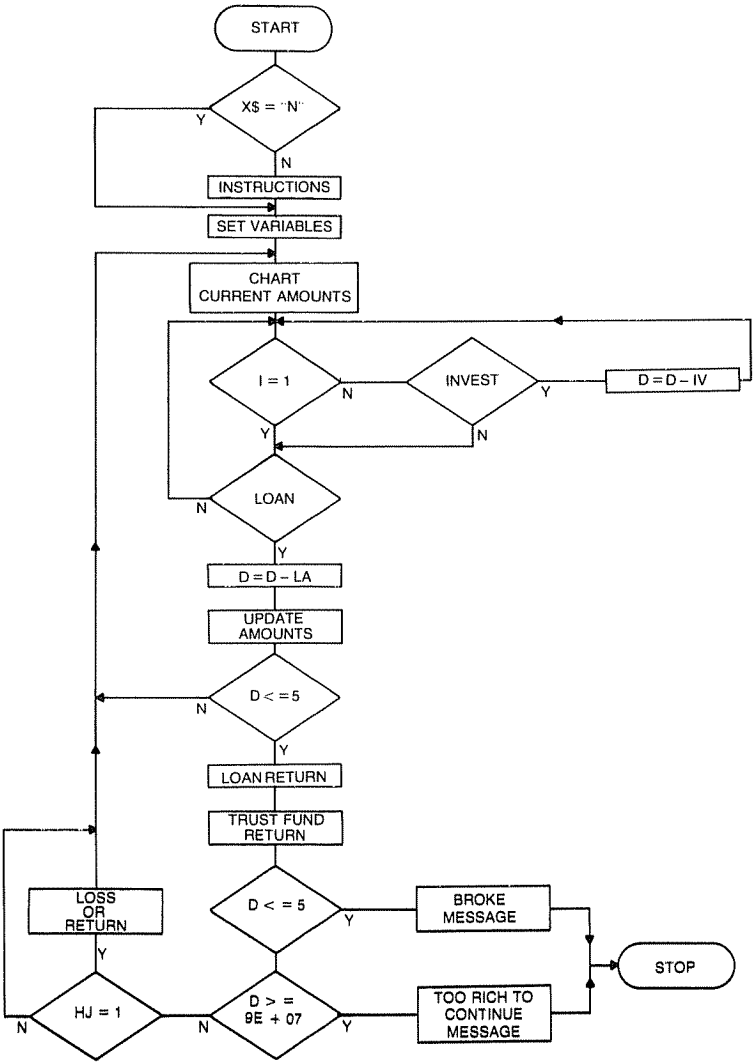


PENNIES OR DOLLARS

Now here is the opportunity you've been longing for: you can get rich, starting with only $50.00. Use your head when you make each loan or investment, and you'll be loaded with computer money. Fail, and you'll end up with less than you started with. You'll be surprised at how fast time flies when the loans start getting paid back. And of course, there are loans that will be defaulted on and investments that will bomb and do nothing else; watch out for these.

Sample Run

```
$$$ PENNIES OR DOLLARS $$$

ALL YOU HAVE TO YOUR NAME IS
  $50.00 DOLLARS. BUT YOU
HAVE COME UP WITH A SCHEME TO
MAKE SOME GOOD MONEY. YOU HAVE
DECIDED TO MAKE LOANS AND TO
INVEST IN SOME PROJECTS. WHO-
EVER YOU MAKE A LOAN TO WILL
NATURALLY HAVE TO PAY IT BACK
WITH INTEREST. AND WHATEVER
YOU INVEST IN COULD MAKE YOU
```



Flowchart for *Pennies or Dollars*.

A MINT...IF YOU MAKE
THE RIGHT INVESTMENTS.

PRESS ENTER?

YOUR INITIALS? D.C.

D.C. YOU'LL HAVE TO BE CAREFUL
HOW YOU LOAN MONEY
OR HOW MUCH YOU INVEST. DON'T
GET GREEDY, OR YOU'LL PROBABLY
LOSE EVERYTHING.

PRESS ENTER TO BEGIN YOUR VENTURE?

Program Listing

```
 10 REM PROGRAM TITLE:
 20 REM PENNIES OR DOLLARS
 30 CLEAR 200:CLS:DIM TA(20),LL(20)
 40 PRINT"$$$ PENNIES OR DOLLARS $$$"
 50 PRINT
 60 INPUT"INSTRUCTIONS REQUIRED (Y/N)";X$
 70 D=50:D$="$$######.##":RANDOM
 80 IF X$="N" THEN 220
 90 PRINT"ALL YOU HAVE TO YOUR NAME IS"
 95 PRINT USING D$;D;
100 PRINT" DOLLARS. BUT YOU"
110 PRINT"HAVE COME UP WITH A SCHEME TO"
120 PRINT"MAKE SOME GOOD MONEY. YOU HAVE"
130 PRINT"DECIDED TO MAKE LOANS AND TO"
140 PRINT"INVEST IN SOME PROJECTS. WHO-"
150 PRINT"EVER YOU MAKE A LOAN TO WILL"
160 PRINT"NATURALLY HAVE TO PAY IT BACK"
170 PRINT"WITH INTEREST. AND WHATEVER"
180 PRINT"YOU INVEST IN COULD MAKE YOU"
190 PRINT"A MINT...IF YOU MAKE"
200 PRINT"THE RIGHT INVESTMENTS."
210 INPUT"PRESS ENTER";X$:CLS
220 INPUT"YOUR INITIALS";I$
225 IF I$="" PRINT"PLEASE, ";:GOTO 220
230 PRINT I$;" YOU'LL HAVE TO BE CAREFUL"
240 PRINT"HOW YOU LOAN MONEY (AMOUNTS)"
250 PRINT"OR HOW MUCH YOU INVEST. DON'T"
260 PRINT"GET GREEDY, OR YOU'LL PROBABLY"
```

```
270 PRINT"LOSE EVERYTHING."
280 PRINT:DIM AI(100),AR(20),AL(20),LA(20)
290 INPUT"PRESS ENTER TO BEGIN YOUR
    VENTURE";X$
300 CLS:R=1:V=129:VV=0
310 GOSUB 320:GOTO 710
320 REM CHART
325 IF ZT=1 THEN 540
330 FOR X=0 TO 127
340 SET(X,4):NEXT:I=1:ZT=1
350 GOSUB 430
360 IF X<>4 THEN X=X+7
370 FOR LT=1 TO LEN(A$)
380 PRINT TAB(X) MID$(A$,LT,1);
390 X=X+1:NEXT
400 I=I+1
410 IF I=6 THEN 540
420 GOTO 350
430 ON I GOTO 440,460,480,500,520
440 A$="ASSETS"
450 X=4:RETURN
460 A$="LOANS"
470 RETURN
480 A$="INVESTED"
490 RETURN
500 A$="RETURNED"
510 RETURN
520 A$="LOST"
530 RETURN
540 IF TH=1 THEN 580:REM AMOUNTS
550 TA(R)=D
560 LL(R)=L:LA(R)=LA
565 IF GK=1 OR IJ=1 THEN RETURN
570 IF M<>0 RETURN
580 X=V:I=1:IV=0
590 GOSUB 645
600 PRINT@X+1,USING D$;T;
610 X=X+13:IF X>=959 THEN GOTO 1660
620 I=I+1
630 IF I>5 RETURN
640 GOTO 590
645 REM AMOUNT FOR T
650 ON I GOTO 660,670,680,690,700
660 IF R=13 AND M=14 THEN T=TA(R+1):RETURN
665 T=TA(R):RETURN
670 IF R=13 AND M=14 THEN T=LL(R+1):RETURN
675 T=LL(R):RETURN
680 T=AI(R):RETURN
```

```
690 T=AR(R):RETURN
700 T=AL(R):RETURN
710 IF D<=6 AND PR=0 THEN 1150
715 PRINT@960,"PRESS ENTER";
720 ZT=0:INPUT X$:IF GK=1 THEN 1680
721 CLS:PR=0
725 IF ZZ=1 GOSUB 1220 ELSE IF ZZ=2 GOSUB 1380
730 REM LOAN/INVESTMENT
740 I=RND(2)
750 ON I GOTO 755,1430
755 PRINT"YOUR ASSETS ARE NOW: ";
760 PRINT TAB(23)USING D$;D
765 PRINT"A LOAN FOR APPROVAL ";I$;":"
770 L=INT(D-RND(10)):IF D<=5 THEN 2100
780 IF L<=3 THEN 770
790 PRINT"LOAN AMOUNT:";TAB(20)USING D$;L
800 P$="#.#%":LA=L:GOSUB 1280
810 PRINT"PERIODIC RATE: ";
820 PRINT TAB(25)USING P$;PE
830 P$="##.#%"
840 PRINT"ANNUAL RATE: ";
850 PRINT TAB(24)USING P$;PE*12
860 REM LENGTH OF PAYBACK (MONTHS)
870 IF L>=1 AND L<=100 THEN PB=12
880 IF L>=101 AND L<=1000 THEN PB=24
890 IF L>=1001 AND L<=9000 THEN PB=36
900 IF L>=9001 AND L<=19000 THEN PB=48
910 IF L>=19001 THEN PB=240
920 REM AMT. BY PERCENT
930 FC=L*PB/100:L=L+FC
940 PRINT"FINANCE CHARGES:";
950 PRINT TAB(20)USING D$;FC
960 PRINT"TOTAL OF LOAN:";
970 PRINT TAB(20)USING D$;L
980 PRINT"PAYMENTS:";
990 PRINT TAB(20)USING D$;L/PB
1000 PRINT
1010 INPUT"APPROVE (Y/N)";A$
1020 IF A$="Y" THEN 1040
1030 GOTO 721
1040 D=D-LA:IF HJ=1 THEN 1800
1050 CLS:IF HJ=0 THEN R=R+1:M=R:GOSUB 540
1060 M=R:R=1:TH=1
1070 GOSUB 320
1080 IF R<M THEN 1100
1090 GOTO 1120
1100 R=R+1:V=V+64
1110 GOTO 1070
```

```
1120 REM PAYBACK
1130 IF D<=6 THEN 1150
1140 V=129:TH=0:GOTO 710
1150 REM LOAN AMOUNT
1160 FOR U=1 TO M
1170 IF LL(U)<>0 THEN 1320
1175 NEXT:PR=1
1180 IF GK=1 RETURN ELSE GOTO 1140
1190 D=D+LL(U):AR=AR+LL(U):GOSUB 2300
1200 LL(U)=0:LA(U)=0:L=0:IF GK=1 THEN 1220
1205 R=R+1:GOSUB 550:GOSUB 1920
1210 ZZ=1:GOTO 1140
1220 PRINT:REM RETURNED
1230 PRINT"LOAN #";RND(10000);"HAS BEEN"
1240 PRINT"PAID IN FULL. AMOUNT ";
1245 PRINT USING D$;AR
1250 PRINT"TOTAL ASSETS NOW:";USING D$;D
1260 FOR W=1 TO 2000:NEXT
1270 AR(R-1)=AR:AR=0:ZZ=0:CLS:RETURN
1280 REM PERCENT
1290 PE=1
1300 PE=PE+RND(.9)
1310 RETURN
1320 REM AMT. LOST
1330 LK=RND(4)
1340 IF LK=1 OR LK=4 THEN 1190
1345 IF D-LA(U)<=6 AND TF=0 THEN 1190
1348 IF U=13 THEN U=14
1350 D=D-LA(U):AL=AL+LA(U)
1360 LL(U)=0:LA(U)=0:L=0:IF GK=1 THEN 1380
1365 R=R+1:GOSUB 550
1370 ZZ=2:GOTO 1130
1380 PRINT:REM LOST
1390 PRINT"LOAN #";RND(10000);"HAS BEEN"
1400 PRINT"DECLARED DEFAULT, SORRY."
1410 PRINT"AMOUNT LOST:";USING D$;AL
1420 GOTO 1250
1430 REM INVESTMENT
1431 IF VV>=101 THEN GOSUB 2200:GOTO 750
1432 IF D<=15 OR R<=2 THEN I=1:GOTO 750
1435 PRINT"A POSSIBLE INVESTMENT..."
1440 IV=RND(5000):IF IV>D+5 THEN 1440
1445 IF D-IV<=10 THEN I=1:IV=0:CLS:GOTO 750
1450 PRINT"YOU CAN NOW MAKE AN INVESTMENT."
1455 IF HJ=1 THEN 1510
1460 PRINT"IF AND WHEN IT PAYS OFF YOU"
1470 PRINT"WILL RECEIVE A % FOR EVERY"
1480 PRINT"LOAN YOU MAKE. IF IT DOES NOT"
```

```
1490 PRINT"PAY OFF YOU WILL LOSE A %"
1500 PRINT"WITH EVERY LOAN YOU MAKE."
1505 PRINT"(PERCENT BASED ON CURRENT RATES)"
1510 PRINT"INVESTMENT #";RND(500)
1520 PRINT"AMOUNT NEEDED:";USING D$;IV
1530 PRINT
1540 INPUT"INVEST (Y/N)";A$
1550 IF A$="Y" THEN VV=VV+1:GOTO 1565
1560 CLS:I=1:IV=0:GOTO 750
1565 IF VV=101 THEN 1431
1570 D=D-IV
1575 AI(VV)=IV:HJ=1
1580 TA(R)=D:GOTO 1560
1590 REM RESET INVEST AMOUNTS
1600 AR=0:AL=0
1610 GOTO 1050
1660 REM RESET VARIABLES
1670 V=129:X=V:GK=1:GOTO 715
1680 CLS:D=TA(R+1):FOR U=1 TO M
1690 IF LL(U)<>0 THEN 1710
1700 GOTO 1720
1710 GOSUB 1160
1720 LL(U)=0:AR(U)=0
1730 AL(U)=0:LA(U)=0:NEXT
1740 R=1:GOSUB 550:GOSUB 1910
1750 GK=0:TH=0:R=1:M=0:GOTO 721
1800 REM INVESTMENT PER %
1810 IJ=1:FOR U=1 TO M:LK=RND(6)
1820 PB=(AI(U)/PE)/ 5
1830 IF AI(U)<>0 THEN 1850
1840 NEXT:IJ=0:R=R+1:GOSUB 1910:GOTO 1590
1850 IF LK=1 OR LK=3 OR LK=6 THEN 1880
1860 D=D+PB:AR=PB:AR(R)=AR
1870 GOSUB 2300:GOTO 1890
1880 D=D-PB:AL=PB:AL(R)=AL
1890 TA(R+1)=D:GOSUB 560
1900 GOTO 1840
1910 REM AMOUNT TO TRUST FUND
1915 IF TF<>0 PRINT"PROCESSING TRUST FUND..."
1920 T$="$$#####,###,###.##"
1925 IF D>=25000 THEN 1940
1930 IF TF<>0 THEN 1955 ELSE RETURN
1940 TF=TF+1000:D=D-1000:PM=1
1950 TD=TD+1000:GOTO 1925
1955 TA(R+1)=D:IF PM=0 OR RF=1 RETURN
1960 PRINT"PROCESSED TO FUND ";USING T$;TD
1965 PRINT"TOTAL AMOUNT OF ";USING T$;TF
1970 PRINT"IS NOW IN A TRUST FUND"
```

```
1980 PRINT"FOR YOUR USE ";I$;"."
1990 FOR W=1 TO 1000:NEXT:PM=0:TD=0:RETURN
2000 REM BUSTED
2010 PRINT:D$="$$###.##"
2020 PRINT"SORRY, YOUR LOANS OR INVESTMENTS"
2030 PRINT"DID NOT PAY OFF FOR YOU ";I$
2040 PRINT"YOUR TOTAL ASSETS ARE TOO LOW"
2050 PRINT"TO CONTINUE (";
2060 PRINT USING D$;D;:PRINT"), SORRY."
2070 PRINT:PRINT"TERMINATION OF RUN..."
2080 END
2100 REM TRUST FUND RETURN
2110 IF TF<>0 THEN 2130
2120 GOTO 2000
2130 D=D+TF:TF=0:RF=1:GOSUB 1920
2140 PRINT
2150 PRINT"AN AMOUNT OF ";USING D$;D
2160 PRINT"IS NOW YOUR TOTAL ASSETS,"
2170 PRINT"DRAWN FROM THE TRUST FUND."
2180 GOSUB 1965
2190 TA(R)=D:RF=0:GOTO 715
2200 REM NO MORE INVESTMENTS
2210 PRINT
2220 PRINT"NO MORE INVESTMENTS CAN BE"
2230 PRINT"MADE NOW ";I$;" SORRY."
2240 FOR W=1 TO 2000:NEXT
2250 I=1:RETURN
2300 REM ASSETS ENOUGH TO RETIRE
2310 IF D>=9E+07 THEN 2330
2320 RETURN
2330 PRINT:IF T$="" THEN 2410
2340 PRINT"YOUR TOTAL ASSETS ARE NOW ENOUGH"
2350 PRINT"FOR YOU TO RETIRE ";I$;"."
2360 PRINT"YOU HAVE A GRAND TOTAL OF ";
2370 PRINT USING T$;D
2380 PRINT
2390 PRINT"THIS PROGRAM MUST BE TERMINATED."
2400 PRINT:GOTO 2070
2410 T$="$$#####,###,###.##"
2420 GOTO 2340
```

Variables and Strings

TA(N) - Total assets
LL(N) - Loans approved
AI(N) - Amount invested
AR(N) - Amount returned from the loan or investment
AL(N) - Amount lost

LA(N) - Loan amount not including interest
D - Dollar amount for TA(N)
D$ & T$ - For printing amounts
I$ - User's initials
R & M - For total amount of assets, loans, and returns
These variables go no higher than 13
W - For Print @ areas
VV - For total investments up to 100 (AI(N)
TH - For print of amounts only
GK & IJ - For return of GOSUB, after amounts have been adjusted
M - For a total of R, print of amounts up to M
IV - Random investment amount
X - For Print @ locations of amounts
ZZ - For messages
LA - Loan amount before interest is tacked on
L - Total loan amount
PE - Interest rates
PB - Payback in months
LK - Random number selector for return or loss of loan or investment
TF - Trust fund

Explanation of the Program Lines

Lines 60-270 display the instructions, if needed, and asks for the user's initials.

Line 300 initializes 3 variables.

Lines 300-700 prints the current chart, which shows the user's total assets, loans, returns, etc.

Line 710 branches to line 1150 for a loan return if D (amount in assets) is less than or equal to $6.00.

Line 720 branches to line 1680 for all loan returns if video has been filled (R=13). At this point, some borrowers could default on their loans.

Line 725 branches to the appropriate print messages for loans returned (ZZ=1) or loans lost (ZZ= 2).

Lines 730-1010 selects and prints a loan amount to be approved or disapproved If I=1. If I= 2, an investment opportunity is offered (line 1430).

Lines 1040-1110, deduct the amount of the loan from the user's total assets and chart the returns.

Lines 1120-1210 control the repayment of a loan if LK is equal to 1 or 4. If it is not (LK= 2 or LK= 3), the loan will be declared

default. Only the principal of the loan will be lost at this point (lines 1335-1420).

Lines 1280-1310 select a random percent for the finance charges.

Lines 1430-1580 select an investment amount (random) if counter VV (total investments) is less than or equal to 100. This investment amount (if approved) will be deducted from the user's total assets, and HJ will then be equal to 1. Each time a loan is made (and HJ=1), all investments from 1 to M (total loans) will either bring a return or a loss, depending on a random integer, LK in lines 1800-1900.

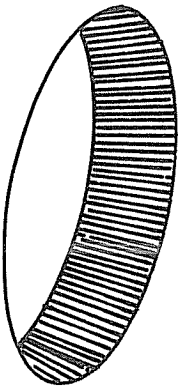
Lines 1910-1990 place an amount in a trust fund for future use if D (total assets) is greater than or equal to 25E+ 03.

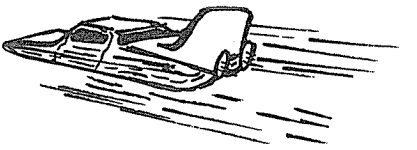
Lines 2000-2080 terminate the program if all loans have been returned, the trust fund is all used, and D<= 5 ($5.00).

Lines 2100-2190 withdraw an amount from the trust fund and adds it to the total assets when needed.

Lines 2200-2250 display a message that no more investments can be made (VV>=101).

Lines 2300-2420 terminate the program if D is greater than or equal to 9E+07.



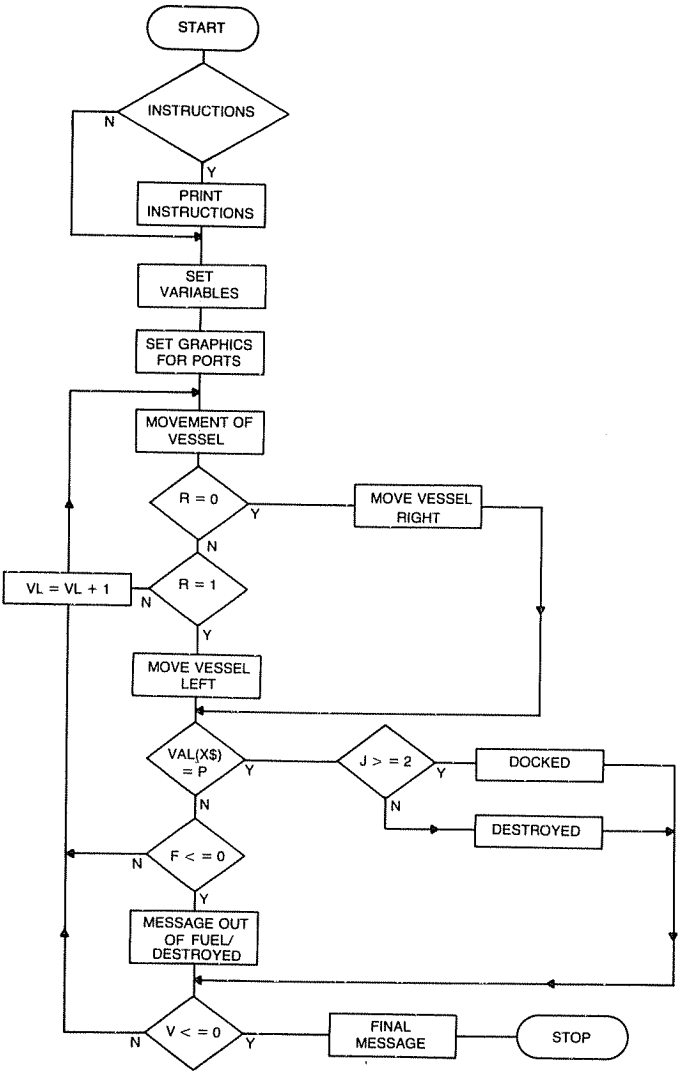


BE PREPARED (PART I)

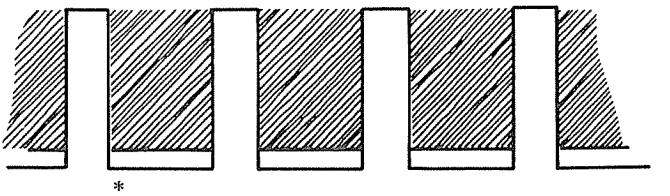
Where is part II you say? Some things must be left for the future, and besides, if you're reading this, you haven't made it through part I yet! At any rate, you are located somewhere in outer space. There are four tubes you must propel your vessels into. Your vessels will move continuously. You will use numeral keys to propel them into these tubes. The first numeral keys that will correctly propel the vessel will be 1 and 2. Each time the vessel makes a cycle (left to right and back to the left) the computer will increase the highest numeral you must press by 1 (i.e., 1–2, 1–3, 1–4, etc., up to 1–9). This means that you will have to guess which numeral will propel the vessel into the tube. Only the right one will do the trick. Refer to pages 1-3 for information on the sound routine. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
BE PREPARED (PART I)

IF YOU HAVEN'T DONE SO, READ AND
FOLLOW ALL INSTRUCTIONS FOR PLAY.
YOU MUST UNDERSTAND EVERYTHING.

INSTRUCTIONS? YES

THIS GAME TAKES
PLACE IN A DISTANT OUTER SPACE
```



Flowchart for *Be Prepared (Part I)*.



FUEL REMAINING: 21 TONS

LOCATION. WHAT YOU WILL SEE IS
FOUR TUBES; THE TOPS ARE CLOSED
WHILE THE BOTTOMS OPEN FOR ENTRY.
WILL BE LOCATED AT
THE BOTTOM OF THE TUBES, IN THE
FORM OF AN ASTERISK. IT WILL BE MOVED
CONTINUOUSLY FROM
LEFT TO RIGHT. NUMERAL KEYS
1-2 WILL PROPEL YOUR CRAFT
INSIDE A TUBE, BUT YOU MUST

PRESS ENTER?

DECIDE WHICH KEY WILL DO THE
JOB. ONLY ONE NUMERAL WILL
WORK, AND THE COMPUTER WILL
INCREASE THE MAXIMUM VALUE WITH EACH
CYCLE. I.E., 1-3, 1-4, ETC. YOU HAVE
20 VESSELS TO DOCK, ONE OTHER
THING THAT CAN STOP YOU IS THE
AMOUNT OF FUEL REMAINING (PER
VESSEL. ONLY IF YOU UNDERSTAND
ALL INSTRUCTIONS, PRESS ENTER TO
DOCK THE FIRST VESSEL?

Program Listing

```
10 REM PROGRAM TITLE: BE PREPARED (PART I)
20 CLS:RANDOM
30 PRINT"BE PREPARED (PART I)
40 PRINT
50 PRINT"IF YOU HAVEN'T DONE SO, READ AND"
60 PRINT"FOLLOW ALL INSTRUCTIONS FOR PLAY."
```

```
 70 PRINT"YOU MUST UNDERSTAND EVERYTHING."
 80 PRINT
 90 INPUT"INSTRUCTIONS";Q$
100 IF MID$(Q$,2,1)<>"E" THEN 400
110 PRINT
120 PRINT"THIS GAME TAKES"
130 PRINT"PLACE IN A DISTANT OUTER SPACE"
140 PRINT"LOCATION. WHAT YOU WILL SEE IS"
150 PRINT"FOUR TUBES; THE TOPS ARE CLOSED"
160 PRINT"WHILE THE BOTTOMS WILL BE OPENED"
170 PRINT"FOR ENTRY."
180 PRINT"YOUR VESSEL WILL BE LOCATED AT"
190 PRINT"THE BOTTOM OF THE TUBES, IN THE"
200 PRINT"FORM OF AN ASTERISK. IT WILL MOVE"
210 PRINT"CONTINIOUSLY FROM"
220 PRINT"LEFT TO RIGHT. NUMERAL KEYS"
230 PRINT"1-2 WILL PROPEL YOUR CRAFT"
240 PRINT"INSIDE A TUBE, BUT YOU MUST"
250 PRINT
260 INPUT"KEY ENTER";X
270 CLS
280 PRINT"DECIDE WHICH KEY WILL DO THE"
290 PRINT"JOB. ONLY ONE NUMERAL WILL"
300 PRINT"WORK, AND THE COMPUTER WILL"
310 PRINT"INCREASE THE MAXIMUM VALUE
    WITH EACH"
320 PRINT"CYCLE. I.E.,1-3, 1-4, ETC.YOU HAVE"
330 PRINT"20 VESSELS TO DOCK. ONE OTHER"
340 PRINT"THING THAT CAN STOP YOU IS THE"
350 PRINT"AMOUNT OF FUEL REMAINING (PER"
360 PRINT"VESSEL). ONLY IF YOU UNDERSTAND"
370 PRINT"ALL INSTRUCTIONS, PRESS ENTER"
380 PRINT"TO DOCK THE FIRST VESSEL";
390 INPUT X
400 CLS
410 REM VESSELS / BEGIN VALUE
420 V=20:DS=0:DK=0:VL=2
430 REM TUBES
440 X=8:Y=26
450 FOR T=Y TO 0 STEP-1
460 SET(X,T):SET(X+5,T):NEXT
470 FOR T=X TO X+5
480 SET(T,0):NEXT
490 REM DO FOUR
500 X=X+31
510 IF X=Y*5+2 THEN 530
520 GOTO 450
530 REM BOTTOM OF PORTS
```

```
 540 W=0:X=8
 550 FOR T=W TO X
 560 SET(T,Y):NEXT:IF P=1 THEN 600
 570 W=X+6:X=X+31
 580 IF W=107 THEN X=127:P=1
 590 GOTO 550
 600 REM FUEL
 610 F=200
 620 F$="FUEL REMAINING:"
 630 X=980
 640 FOR I=1 TO LEN(F$)
 650 PRINT@X,MID$(F$,I,1);
 660 X=X+1:GOSUB 1500
 670 NEXT:IF G=1 RETURN
 680 Q=X+1
 690 PRINT@Q,F;
 700 REM START LOCAL
 710 L=640:V$="*":ZX=255:PK=1
 720 M=L+63:P=RND(VL):R=0
 730 PRINT@L,V$;
 740 REM BY INKEY$
 750 X$=INKEY$
 760 IF VAL(X$)=P THEN 900
 770 FOR T=1 TO 5:NEXT
 780 PRINT@L," ";
 790 IF R=1 THEN 810
 800 L=L+1:GOTO 840
 810 L=L-1
 820 IF L<=M THEN 720
 830 GOTO 850
 840 IF L>=M THEN M=L-63:R=1:VL=VL+1
 850 F=F-1:IF VL>=9 THEN VL=9
 860 IF F<=0 THEN 1200
 870 PRINT@Q,F;:PRINT@Q+5,"TONS";
 880 IF INT(F/10)=F/10 THEN 1540
 885 IF F<=10 THEN 1540
 890 GOTO 730
 900 REM PORT?
 910 J=1:ZX=50:PK=0
 920 PRINT@L," ";
 930 L=L-64
 940 PRINT@L,V$;
 950 J=J+1:GOTO 1540
 960 IF PEEK(L+15360-64)<>32 THEN 990
 970 GOTO 920
 990 IF J<=2 THEN 1030
1000 F$=" VESSEL    DOCKED    SAFELY "
1010 G=1:GOSUB 630
```

```
1020 DK=DK+1:GOTO 1070
1030 F$="V E S S E L   D E S T R O Y E D"
1040 FOR T=1 TO 500:NEXT:PRINT@L," ";
1050 G=1:X=975:GOSUB 640
1060 DS=DS+1
1070 GOSUB 1600
1080 F$=STRING$(32,32):X=975
1090 GOSUB 640
1100 V=V-1:G=0
1110 IF V<=0 THEN 1300
1120 VL=VL+1
1130 IF VL>=9 THEN VL=9
1140 GOTO 600
1200 REM FUEL EXHAUSTED
1210 F$="F U E L       E X H A U S T E D"
1220 G=1:X=975
1230 GOSUB 640
1240 FOR T=1 TO 500:NEXT
1250 GOTO 1030
1300 REM ALL VESSELS IN
1310 G=1:XX=0
1320 F$="A L L     V E S S E L S     I N"
1330 X=975:GOSUB 640:GOSUB 1600:X=XX:CLS
1340 F$="YOU HAVE DOCKED"
1350 GOSUB 640:PRINT DK;:F$="VESSELS"
1355 X=X+4
1360 GOSUB 640:X=XX+64:XX=XX+64
1370 F$="YOU HAVE DESTROYED"
1380 GOSUB 640:PRINT DS;:F$="VESSELS"
1385 X=X+3
1390 GOSUB 640:X=XX+64:XX=XX+64
1400 F$="END OF MISSION..."
1410 GOSUB 640
1420 END
1500 REM SOUND ROUTINE
1510 POKE 31266,ASC(MID$(F$,I,1))+10
1520 Z=USR(0)
1530 RETURN
1540 REM MORE SOUND
1550 POKE 31266,ZX:Z=USR(0)
1560 IF PK=1 THEN 1580
1570 GOTO 960
1580 IF F<=100 THEN ZX=ZX-10
1590 GOTO 890
1600 REM FOR MESSAGE LOOP
1610 FOR T=1 TO 1200:NEXT
1620 RETURN
```

Variables and Strings

V - Number of vessels
DS - Vessels destroyed (counter)
DK - Vessels docked (counter)
VL - The beginning value for a random number between 1 and 2. This value will increase up to 9 in increments of 1 for each cycle (cycle: left to right then back to left) and each time a craft is safely docked X,Y, and W — For graphics of tubes (docks)
F - Fuel amount
F$ - Contents of different messages, depending on area of flow
X - Used again for Print @ locations of messages
Q - Print @ area of F (fuel)
G - Simple 0/1 true/false variable for messages
L - Start location of vessel
V$ - Vessel (asterisk)
ZX - Used in conjunction with sound output
PK - For GOTO line definition
M - Limit of L for asterisk movement
P - Random selection of VL
K - For left (R = 0) or right (R=1) movement of vessel
J - To note if vessel has docked safely (J>= 3)
Z - For USR call of sound routine

Explanation of the Program Lines

Lines 10-400 display the general playing instructions.

Line 420 initializes several of the variables.

Lines 430-590 display the graphics for the tubes (ports) in which each vessel must dock.

Lines 600-620 set the amount of fuel and one of the messages for F$.

Lines 630-690 print the different messages in F$. If G is equal to 1, lines 630 or 640 through lines 630-670 are being used as from another part of the program.

Lines 700-890 form the main part of the program. Movement of the asterisks will continue left to right and then right to left until the VAL of X$ is equal to P, or all of the fuel is exhausted.

Lines 900-970 move the vessel upward until the condition at 960 is met.

Line 990 tests the value of J. If it is less than or equal to 2 the vessel is considered destroyed.

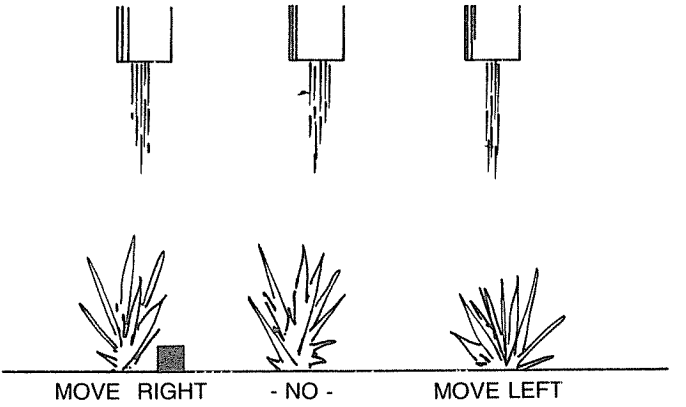
Lines 1000-1250 print the appropriate message indicating the

status of the vessel, advance the counters, and continue the program if V>=1.

Lines 1300-1420 display messages indicating the number of vessels docked and/or destroyed and then terminate the program.

Lines 1500-1590 is the routine for the sound output used when the status messages are displayed.

Lines 1600-1620 form a time loop (pause) between appropriate messages.

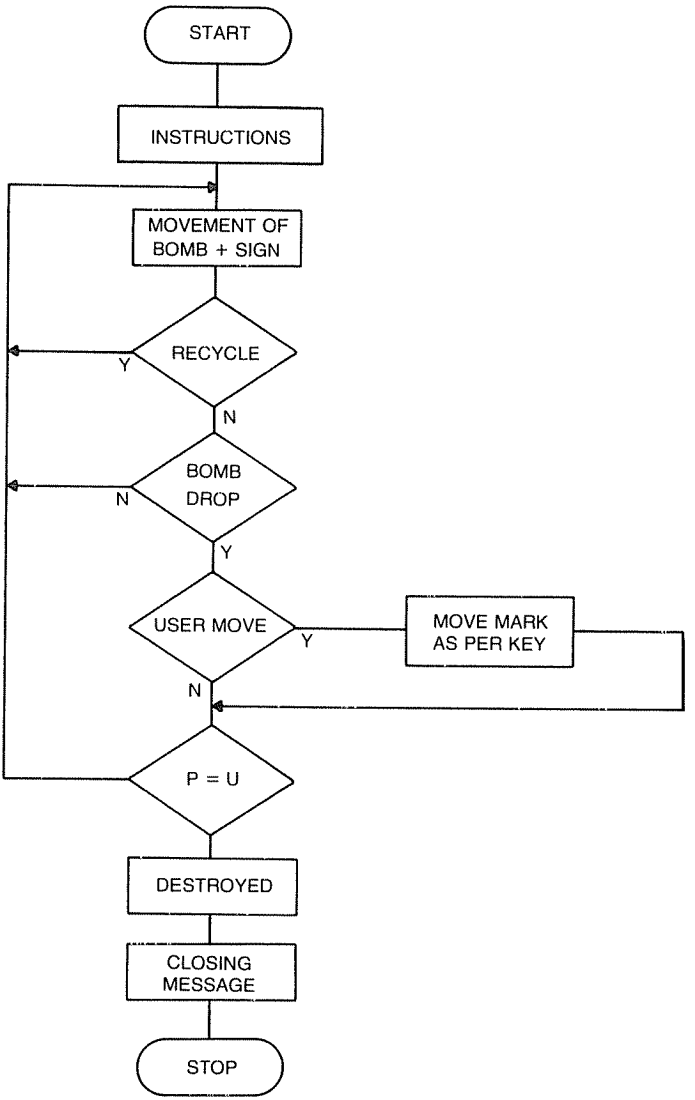


RAIN OF TERROR

From the sky bombs will fall on your exact location. You will use the left and right arrow keys to move out of the path of destruction, but you will have to be fast! Keep an eye on the POINTS BEFORE HIT message to obtain the highest score before termination. Your score depends on when you move out of the path of the bomb. The longer you wait, the higher your score. Remember to use the proper value for SL in line 30. Refer to page 3. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
***RAIN OF TERROR***

INSTRUCTIONS (Y/N)? YES

YOU WILL BE BOMBARDED FROM THE
SKY, BY FALLING BOMBS. ALL YOU
NEED DO TO STAY ALIVE IS TO
USE THE LEFT ARROW AND THE
RIGHT ARROW KEYS TO GET
OUT OF THE WAY.
IT SOUNDS VERY
SIMPLE, BUT YOU MUST BE FAST!

PRESS A KEY TO BEGIN
```



Flowchart for *Rain of Terror*.

Program Listing

```
 10 REM PROGRAM TITLE: RAIN OF TERROR
 20 CLEAR 200:RANDOM
 30 CLS:PRINT:SL=31266
 40 PRINT"*** RAIN OF TERROR ***"
 50 PRINT:X$=STRING$(63,32)
 60 INPUT"INSTRUCTIONS (Y/N)";A$
 70 IF A$="N" THEN 190
 80 PRINT"YOU WILL BE BOMBARDED FROM THE"
 90 PRINT"SKY, BY FALLING BOMBS. ALL YOU"
100 PRINT"NEED DO TO STAY ALIVE IS TO"
110 PRINT"USE THE LEFT ARROW AND THE"
120 PRINT"RIGHT ARROW KEYS, TO GET"
130 PRINT"OUT FROM UNDER THE FALLING BOMBS."
140 PRINT"IT ALL SOUNDS VERY"
150 PRINT"SIMPLE, BUT YOU MUST BE FAST !!"
160 PRINT
170 PRINT"PRESS A KEY TO BEGIN"
180 A$=INKEY$:IF A$="" THEN 180
190 CLS
200 REM BORDER / MESSAGES
210 PRINT@0,STRING$(63,191);
220 I$=CHR$(143):GOSUB 510
230 R=0:REM POINTS
240 P=64:PP=P+62:U=P+896:UV=U:UW=U+61:RR=0
250 PRINT@P,"+";:PRINT@U,"  ";:P=P+2
260 U=U+2:PRINT@P-2," ";:PRINT@U,I$;
270 IF P>=PP THEN PRINT@U," ";:GOTO 240
280 POKE SL,40:D=USR(0):PX=RND(127)
290 IF PX<=63 THEN 280
300 IF ABS(P-PX)<=8 THEN 320
310 GOTO 250
320 REM A DROP
330 PL=P
340 PRINT@P,"*";:P=P+64
350 PRINT@P-64,"  ";
360 IF P>PL+896 THEN 410
370 K=PEEK(14440):IF K=32 THEN 420
380 IF K=64 THEN 470
390 R=R+25:RR=RR+25:PRINT @18,RR;:PRINT@55,R;
400 IF P=U THEN 550 ELSE GOTO 340
410 PRINT @960,X$;:P=PL:RR=0:GOTO
420 PRINT@U," ";
430 U=U-2
440 IF U<=UV THEN 480
450 PRINT@U,I$;
460 U=PL+896:GOTO 340
```

```
470 PRINT@U," ";
480 U=U+2
490 IF U>=UW THEN U=UV
500 GOTO 450
510 REM MESSAGES
520 PRINT@0,"POINTS BEFORE HIT";
530 PRINT@40,"TOTAL POINTS";
540 RETURN
550 REM BOMBED
560 R=R-RR:PRINT@55,R;:PRINT@960,X$;
570 IF U<=962 THEN U=U+3
580 IF U>=1020 THEN U=U-4
590 U1=U-64:U2=U1-1:U3=U1+1:U4=1
600 PRINT@U,"*";
605 POKE SL,200:D=USR(0)
610 PRINT @U1,"*";:PRINT@U2,"*";
615 POKE SL,230:D=USR(0)
620 PRINT@U3,"*";
625 POKE SL,240:D=USR(0)
630 FOR T=1 TO 200:NEXT:U4=U4+1
640 PRINT@U," ";:PRINT@U2,STRING$(10,32);
650 IF ABS(U-U1)>=256 THEN 670
660 U1=U1-64:U2=U1-U4:U3=U1+U4:GOTO 600
670 REM GONE
680 FOR T=1 TO 1000:NEXT
690 CLS
700 PRINT"YOU HAVE BEEN DESTROYED BY ONE"
710 PRINT"OF THE FALLING BOMBS. YOUR"
720 PRINT"TOTAL POINTS ACCRUED BEFORE"
730 PRINT"YOU WERE DESTROYED WERE:";R
740 PRINT:FOR T=1 TO 1000:NEXT
750 A$="END OF RAIN OF TERROR."
760 FOR I=1 TO LEN(A$)
770 PRINT MID$(A$,I,1);:POKE SL,30
780 D=USR(0)
790 NEXT
800 END
```

Variables and Strings

SL - Sound location (machine language routine)
X$ - For clearing area on video
I$ - User's mark
R - Points
P,PP,U,UV,UW - All used in conjunction with one another for bomb location before drop and user's location
D - USR call for output of sound

PX - Used for a bomb drop, if less than or equal to 8
PL - Length of the drop
K — User's movement (peek)
RR - Points before actually hit by bomb
U1,U2,U3,U4 - For explosion of user's mark when hit
T - Time loop (pause)

Explanation of the Program Lines

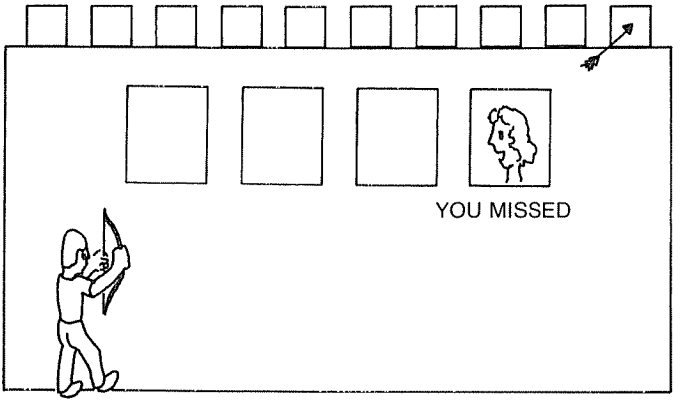
Lines 10-190 initialize the sound location variable and display the general instructions.

Lines 200-310 initialize more variables and keep the bomb and location of the drop moving before a drop.

Lines 320-540 drop a bomb on the user's location, determine the user's location (if any), increase the score until the hit occurs, and determine the total score. If P= U (line 400), the user is destroyed.

Lines 550-660 display the explosion of the user's mark when it is hit by a bomb.

Lines 670-800 display the user's total number of points obtained before the mark was destroyed.

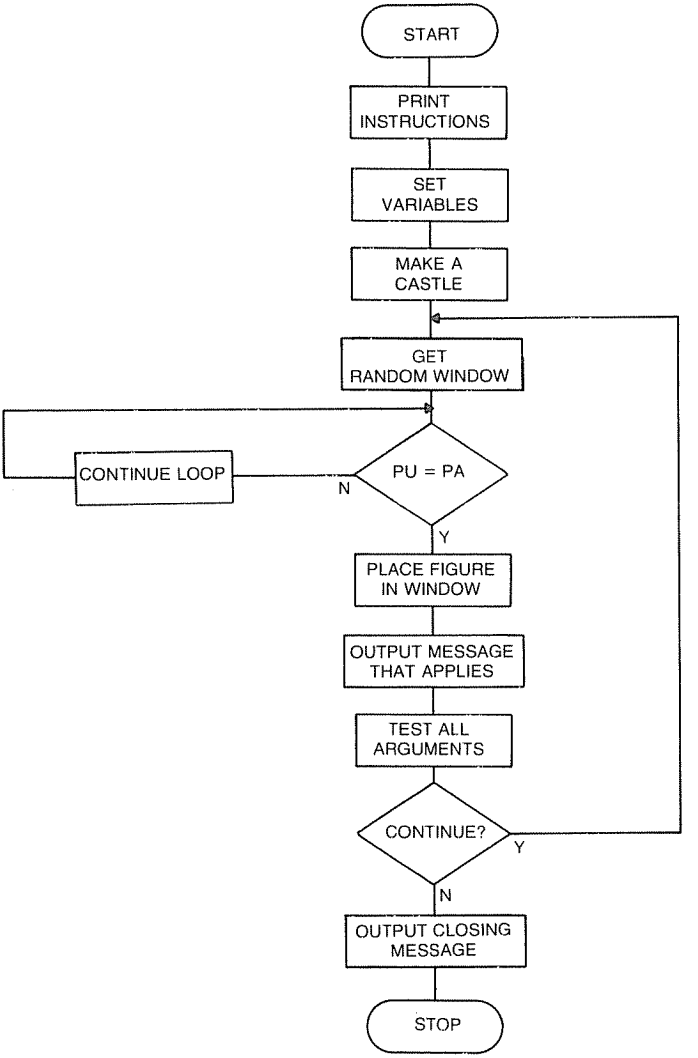


KNIGHTS

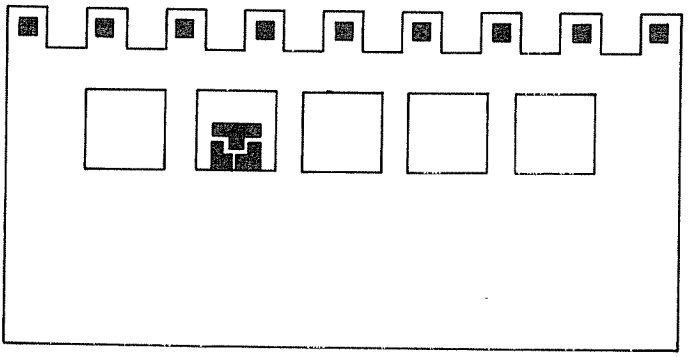
You've been hired by the king to destroy some knights that have escaped from the dungeon and gone into a castle. You are a sharpshooter, but it won't be that easy. The knights have captured some hostages, so think before you fire into that open window; you could shoot a hostage. If you destroy all of the hostages, the king will have your head! Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
>>>>KNIGHTS<<<<

INSTRUCTIONS NEEDED? YES

THERE ARE 10 KNIGHTS IN A CASTLE.
THERE ARE KNIGHTS THAT HAVE DONE
WRONG AND WERE IN A DUNGEON
WAITING TO HAVE THEIR HEADS
REMOVED. THEY HAVE BROKEN OUT OF
THE DUNGEON AND GONE INTO A
CASTLE. ALONG WITH THEM ARE
8 HOSTAGES. THE KING
WILL NOT MEET TO THE CONDITIONS
THEY HAVE SET, SO THEY'RE
GOING TO STAY UNTIL THE END...
```



Flowchart for *Knights*.



YOU HAVE 9 ARROWS TO FIRE.

PRESS ENTER TO RESUME?

YOU ARE THE SHARPSHOOTER SELECTED
BY THE KING TO TERMINATE EACH
OF THESE KNIGHTS. YOU ARE
EQUIPPED WITH A HIGH-POWER CROSS-
BOW. ALONG THE CASTLE WALLS ARE
FIVE WINDOWS. EACH TIME A FIGURE
APPEARS AT A WINDOW YOU'LL HAVE
THE OPTION OF WHETHER OR
NOT TO FIRE (BY HITTING
THE SPACE BAR). THE
FIGURE THAT APPEARS AT A WINDOW
COULD BE ONE OF THE HOSTAGES.
IF IT IS A KNIGHT AND YOU DON'T
FIRE, YOU COULD BE HIT...

PRESS ENTER TO BEGIN?

Program Listing

```
10 REM PROGRAM TITLE: KNIGHTS
20 CLEAR 700
30 CLS:RANDOM
40 PRINT")))) KNIGHTS ((((
50 PRINT
60 PRINT"INSTRUCTIONS NEEDED";
70 INPUT T$
80 IF T$="N" OR T$="NO" THEN 370
```

```
 90 PRINT"THERE ARE 10 KNIGHTS IN A CASTLE"
100 PRINT"THESE ARE KNIGHTS THAT HAVE DONE"
110 PRINT"WRONG AND WERE IN A DUNGEON"
120 PRINT"WAITING TO HAVE THEIR HEADS"
130 PRINT"REMOVED. THEY HAVE BROKEN OUT OF"
140 PRINT"THE DUNGEON AND GONE INTO A"
150 PRINT"CASTLE TAKING EIGHT HOSTAGES"
160 PRINT"WITH THEM. THE KING WILL"
170 PRINT"NOT AGREE TO THE CONDITIONS"
180 PRINT"THEY HAVE SET DOWN, SO THEY'RE"
190 PRINT"GOING TO STAY 'TIL THE END..."
200 PRINT:INPUT"PRESS ENTER TO RESUME";T$
210 PRINT@0,CHR$(30)
220 PRINT"YOU ARE THE SHARPSHOOTER SELECTED"
230 PRINT"BY THE KING TO TERMINATE EACH"
240 PRINT"OF THESE KNIGHTS. YOU ARE"
250 PRINT"EQUIPPED WITH A HIGH-POWER CROSS-"
260 PRINT"BOW. ALONG THE CASTLE WALLS ARE"
270 PRINT"FIVE WINDOWS. EACH TIME A FIGURE"
280 PRINT"APPEARS AT A WINDOW YOU'LL HAVE"
290 PRINT"THE OPTION OF WHETHER OR NOT TO FIRE"
300 PRINT"(BY HITTING THE SPACE BAR). THE"
310 PRINT"FIGURE THAT APPEARS AT A WINDOW"
320 PRINT"COULD BE ONE OF THE HOSTAGES."
330 PRINT"IF IT IS A KNIGHT AND YOU DON'T"
340 PRINT"FIRE, YOU COULD BE HIT..."
350 PRINT
360 INPUT"PRESS ENTER TO BEGIN";T$
370 CLS:K=10:HD=8:CR=10
380 REM MAKE A CASTLE
390 C$=STRING$(62,191)
400 A=320
410 PRINT@A,C$;
420 IF A>=768 THEN 440
430 A=A+64:GOTO 410
440 REM TOP OF CASTLE
450 A=256:C$=CHR$(183)+CHR$(187)
460 PRINT@A,C$;
470 A=A+4
480 IF A>=320 THEN 500
490 GOTO 460
500 REM NOW FIVE WINDOWS
510 C$=STRING$(5,32):J=1
520 A=A+72:XX=19
530 PRINT@A,C$;:PRINT@A+64,C$;
540 A(J)=A+65:J=J+1
550 IF A>=432 THEN 570
560 A=A+10:GOTO 530
```

```
 570 REM FIGURE
 580 F$=CHR$(182)+CHR$(143)+CHR$(185)
 590 REM GET A WINDOW
 600 W=RND(5)
 610 WI=A(W)
 620 REM PAUSE BEFORE PLACING
 630 PA=RND(700):IF PA<=250 THEN 630
 640 FOR PU=1 TO 700
 650 IF PU=PA THEN 670
 660 NEXT:GOTO 630
 670 PRINT@WI,F$;
 680 REM SHARPSHOOTER TO FIRE
 690 FOR TI=1 TO 100
 700 W$=INKEY$
 710 IF W$<>"" THEN 730
 720 NEXT
 730 REM KNIGHT OR HOSTAGE
 740 H1=RND(5):H2=RND(5):H3=RND(5)
 750 H=ABS(H1-H2-H3)
 760 PRINT@WI,STRING$(3,32);
 762 IF H=W THEN 1240
 765 IF HD<=0 THEN 840
 770 IF H<=0 OR H>=5 AND TI<=99 THEN 790
 780 GOTO 840
 790 REM A HOSTAGE
 800 PRINT@0,;:IF TI>=75 THEN 880
 810 PRINT"YOU HAVE JUST KILLED ONE"
 820 PRINT"OF THE HOSTAGES, TURKEY !!"
 830 GOTO 970
 840 PRINT@0,;:IF TI>=100 THEN 880
 850 PRINT"GOOD SHOOTING, YOU HAVE"
 860 PRINT"KILLED ANOTHER KNIGHT."
 870 GOTO 940
 880 PRINT@0,;:IF H>=0 AND H<=2 THEN 1020
 890 PRINT"THINK FASTER !! YOU HAVE BEEN"
 900 PRINT"HIT BY ONE OF THE KNIGHTS !!"
 910 HK=HK+1
 920 IF HK>=6 THEN 1060
 930 GOTO 990
 940 K=K-1:PRINT K;"KNIGHTS REMAIN."
 950 IF K<=0 THEN 1120
 960 GOTO 990
 970 HD=HD-1
 980 IF HD<=0 THEN 1180
 990 FOR TX=1 TO 1500:NEXT
1000 REM CLEAR PRINT AREA
1010 PRINT@0,STRING$(255,32);:GOTO 1400
1020 IF HD<=0 THEN 890
```

```
1030 PRINT"GOOD THING YOU DID NOT FIRE"
1040 PRINT"THAT WAS A HOSTAGE !!"
1050 GOTO 990
1060 REM HIT ENOUGH
1070 PRINT@0,;
1080 PRINT"YOU HAVE BEEN HIT";HK;"TIMES."
1090 PRINT"YOUR BODY CANNOT STAND"
1100 PRINT"ANYMORE. YOU ARE DYING....."
1110 GOTO 1530
1120 REM KNIGHTS OMITTED
1130 PRINT@0,;
1140 PRINT"YOU HAVE KILLED ALL 10"
1150 PRINT"KNIGHTS THE KING WILL BE"
1160 PRINT"PROUD OF YOU MY FRIEND....."
1170 GOTO 1530
1180 REM HOSTAGES GONE
1190 PRINT@0,;
1200 PRINT"YOU HAVE KILLED ALL OF THE"
1210 PRINT"HOSTAGES...YOU TURKEY !!"
1220 PRINT"THE KING WILL HAVE YOUR HEAD !!"
1230 GOTO 990
1240 REM COMPLETE MISS
1245 IF TI)=100 THEN 765
1250 XX=RND(122)
1260 IF XX(=2 THEN 1260
1270 YY=RND(35)
1280 IF YY(=17 THEN 1270
1290 RESET(XX,YY)
1300 PRINT@0,;
1310 PRINT"YOU'LL HAVE TO DO BETTER"
1320 PRINT"THAN THAT, YOU'VE MISSED !!"
1330 GOTO 990
1350 REM CROSSBOW ROUNDS
1360 PRINT@910,;
1370 PRINT"YOU HAVE";CR;"ARROWS TO FIRE."
1380 RETURN
1400 REM ENOUGH ARROWS LEFT
1410 IF TI(=99 THEN CR=CR-1
1420 GOSUB 1350:IF CR(=0 THEN 1440
1430 GOTO 590
1440 REM END
1450 FOR TX=1 TO 2000:NEXT
1460 CLS
1470 PRINT"YOU HAVE EXHAUSTED YOUR"
1480 PRINT"SUPPLY OF CROSSBOW ARROWS."
1490 IF K()0 THEN 1510
1500 GOTO 1530
1510 PRINT"THERE ARE STILL";K;"KNIGHTS"
```

```
1520 PRINT"IN THE CASTLE....."
1530 PRINT
1540 IF HD<=0 THEN 1560
1550 GOTO 1600
1560 PRINT"YOU ARE A BLUNDERING SHARP-"
1570 PRINT"SHOOTER, YOU IDIOT !!"
1580 PRINT"YOU HAVE HIT AND KILLED "
1590 PRINT"THE 8 HOSTAGES !!!"
1600 PRINT
1610 PRINT"END OF RUN."
1620 END
```

Variables and Strings

K - Number of knights
HD - Total number of hostages
CR - Total arrows
C$ - CHR$ for castle and windows
A - Print @ location for structure of castle
A(J) - Locations of windows (5)
W - Random window selection
WI - Window select, A(W)
PA - Pause before placing figure in selected window
W$ - INKEY$ to fire at figure
H1,H2,H3,H - Used to determine if the figure was a hostage
TI - Time loop for the length of time figure will appear in window
HK - Counter user hit by knight
XX,XY - Used to reset an area of the castle for a user miss

Explanation of the Program Lines

Lines 10-360 display the general operating instructions.

Line 370 initializes 3 variables.

Lines 380-560 display a castle and set the 5 window locations to the subscripted variable A(J).

Lines 570-580 establish the figure that will appear in a window.

Lines 590-610 select (randomly) which one of the five windows the figure will appear in.

Lines 620-660 establish a random-length pause determined by the variable PA, before the figure is printed.

Lines 680-720 use the INKEY$ function for the user to fire upon the figure.

Lines 730-750 use variables to determine whether figure was a hostage or a knight.

Line 760 clears the figure from the window.

Line 762 determines whether or not the shot missed completely.

Line 765 sends program control to line 840 where a knight eliminated if TI<= 99.

Line 770 determines whether or not a hostage hit, if TI<= 99.

Line 780 is the same as line 762 and is used if previous conditions were not met.

Lines 790-830 display the hostage killed message.

Lines 840-870 display the knight killed message.

Lines 880-930 indicate that the user has been hit by a knight. If the user receives at least 6 hits, he or she will be declared dead (line 920).

Lines 940-960 display a message indicating how many knights remain. If K<= 0 all knights have been eliminated.

Line 950 sends program control to line 1120 if all the knights are dead.

Lines 970-980 are accessed when a hostage is killed. If all hostages are killed, the king will have the user's head (line 980).

Lines 990-1010 form a delay loop and clear the print messages.

Lines 1020-1050 print a message indicating that a hostage was in window.

Lines 1060-1110 display the message that the user is dying.

Lines 1120-1170 display the message that all the knights have been killed.

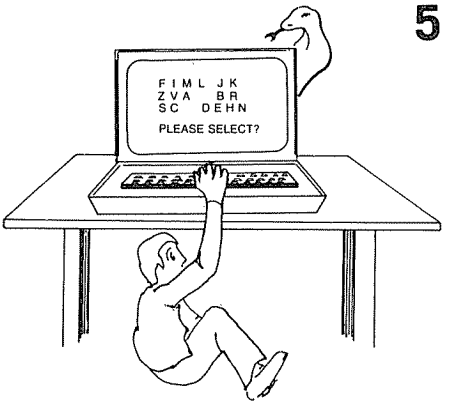
Lines 1180-1230 display the message that all the hostages have been killed, but the program continues so that the user can finish off the knights.

Lines 1240-1330 are used when the user misses completely. Message is displayed, and an area of the castle is reset to show what area was hit.

Lines 1350-1380 display a message that indicates how many arrows remain.

Lines 1400-1430 test the number of arrows remaining. If there aren't any, the program is terminated.

Lines 1440-1620 terminate the program and display the appropriate messages.



Games to Keep You Alert or Put You to Sleep

Triangle is a game of chance that tests your luck in avoiding landing on the spot chosen by the computer.

Mind Invasion forces you to move quickly to avoid being crushed.

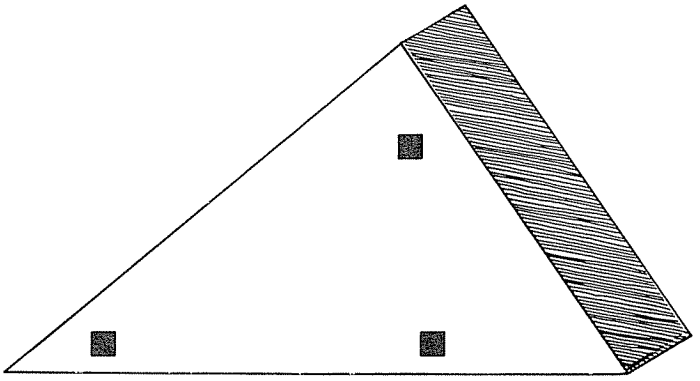
Trapped challenges you to trap the computer's piece. Watch out—it can burn you.

Elevator tests your memory. Can you get all the customers to the floors they want to go to?

Farewell offers a merciful death. All you must do is land on the spot where the cobra is hiding. To win you must land on the spot indicated by a quick flash.

Insomnia will put you to sleep. All you have to do is watch the screen.

Rainy Daze puts your attention span on trial. How long can you concentrate on the numbers on the screen?



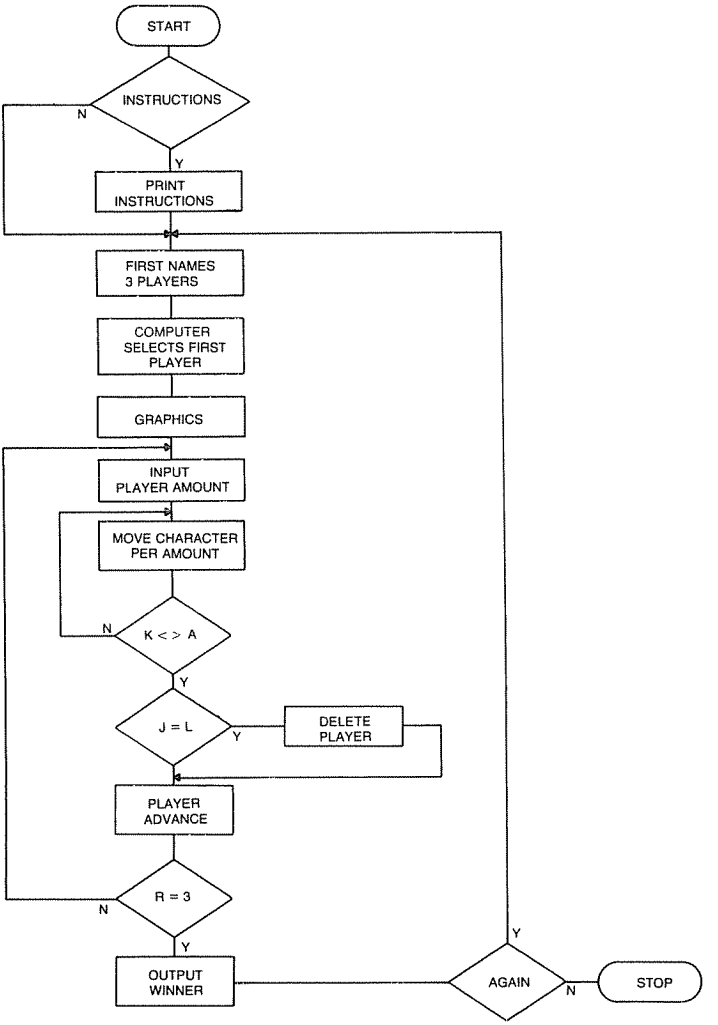
YOU'RE OUT OF THE GAME

TRIANGLE

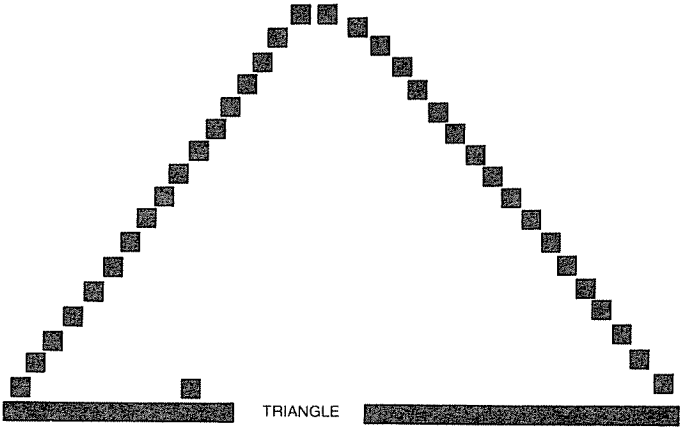
Moving around in a triangle. You and two other people will try tc avoid landing on the spot selected by the computer. There is a way to come out the victor almost every time. That is all I will tell you; you'll have to figure out how yourself. Don't get your arms tangled with the other two players during the trips around the triangle! Refer to pages 1-3 for information on the sound routine.

Sample Run

```
...TRIANGLE...

OUTLINE FOR PLAY?YES
THIS PROGRAM REQUIRES 3 PLAYERS.
THE COMPUTER WILL SELECT A LOCATION
AT THE START OF THE PLAY. EACH
PLAYER IN TURN WILL ENTER
A NUMBER FROM 1 TO 9. THE MOVE
AROUND THE TRIANGLE WILL
BEGIN. IF AT ANYTIME A PLAYER
LANDS ON THE SPOT THE COMPUTER
HAS CHOSEN, THAT PLAYER WILL BE
ELIMINATED FROM THE GAME.

PRESS A KEY...

THIS WILL CONTINUE UNTIL THERE
```



Flowchart for *Triangle*.



TRIANGLE

IS ONLY ONE PLAYER LEFT, THE
WINNER. THINK BEFORE YOU ENTER A
NUMBER AND KEEP YOUR EYE
ON THE OPPONENT.
3 FIRST NAMES, PLEASE.

Program Listing

```
 10 REM PROGRAM TITLE: TRIANGLE
 20 REM REQUIRES THREE PLAYERS
 30 CLS
 40 PRINT"   TRIANGLE   "
 50 PRINT
 60 INPUT"OUTLINE FOR PLAY";X$
 70 IF MID$(X$,2,1)()"E" THEN 240
 80 PRINT"THIS PROGRAM REQUIRES 3 PLAYERS."
 90 PRINT"EACH PLAYER WILL TRY TO EVADE"
100 PRINT"THE OTHER, IN A NEVER ENDING"
110 PRINT"TRIANGLE  MORE LIKE A CHASE,"
120 PRINT"THE SECOND PLAYER WILL MOVE"
130 PRINT"AROUND THE TRIANGLE, SEARCHING"
140 PRINT"FOR THE FIRST AND THIRD. THE THIRD"
150 PRINT"PLAYER WILL SEARCH FOR THE SECOND"
160 PRINT"OR FIRST. THEN THE FIRST WILL"
170 PRINT"SEARCH FOR THE SECOND OR THIRD..."
180 PRINT"!!! W H E W !!!"
190 GOSUB 1000:CLS
200 PRINT"THIS WILL CONTINUE UNTIL ONLY ONE"
210 PRINT"PLAYER IS LEFT, THAT PLAYER WILL"
```

```
220 PRINT"BE THE WINNER. MOVE AMOUNTS WILL"
230 PRINT"RANGE FROM 1 THROUGH 9. ENTER NOW"
240 PRINT"THE THREE PLAYERS FIRST NAMES";
250 FOR I=1TO3
260 INPUT N$(I):NEXT:Y=2
270 PRINT"FIRST PLAYER SELECTION..."
280 FOR I=1TO10:L(I)=I:NEXT:FOR T=1TO1000
290 NEXT:R=1:GOSUB 1400:A=RND(3)
300 IF A=1 THEN 360
310 IF A=2 THEN 340
320 N$(0)=N$(3).FOR T=ATO1 STEP-1
330 N$(T)=N$(Y):Y=Y-1:NEXT:GOTO 350
340 N$(0)=N$(2):N$(2)=N$(3):N$(3)=N$(1)
350 N$(1)=N$(0)
360 I=1:RANDOM
370 PRINT"THE FIRST PLAYER IS: ";N$(I)
380 PRINT
390 GOSUB 1000:CLS
400 REM GRAPHICS WITH SOUND
410 POKE 16526,0:POKE 16527,122
420 W=897
430 PRINT@W,CHR$(143);
440 POKE 31265,30:POKE 31266,100:X=USR(0)
450 IF W)=958 THEN 470
460 W=W+1:GOTO 430
470 PRINT@923," TRIANGLE ";:M=2:N=124:Q=41
480 SET(M,Q).SET(M+1,Q):SET(N,Q):SET(N+1,Q)
490 POKE 31265,30:POKE 31266,N:X=USR(0)
500 M=M+2:N=N-2:Q=Q-1
510 IF ABS(M-Q)=54 THEN 530
520 GOTO 480
530 N$=N$(I):PRINT@0,N$;" YOUR LOCATION...  ";
540 F=0:IF P=1 THEN 560
550 J=863:REM MOVEMENT TO LEFT
560 PRINT@J,CHR$(143);
570 IF F()3 THEN 590
580 GOTO 630
590 F=F+1
600 POKE 31265,30:POKE 31266,200:X=USR(0)
610 FOR T=1TO100.NEXT:PRINT@J,CHR$(32);
620 FOR T=1TO125:NEXT:GOTO 560
630 REM ENTRY AMOUNT
640 PRINT@0,;:NN=220
650 PRINT N$;" ENTER A NUMBER?";
660 GOSUB 1010:A=VAL(X$):K=0:GOSUB 990
670 IF K()A THEN 690
680 GOTO 940
690 GOSUB 1100
```

```
700 J=J-1:GOSUB 1130
710 IF D=1 THEN GOSUB 920:GOTO 740
720 IF J<=837 THEN GOSUB 900
730 IF J=350 THEN GOSUB 1100:GOTO 780
735 IF J=352 THEN GOSUB 1100:GOTO 780
740 IF J=889 THEN D=0:NN=220
750 K=K+1:GOTO 1200
760 POKE 31265,30:POKE 31266,NN:X=USR(0)
770 GOTO670
780 J=J+3:D=1
790 GOTO 740
900 GOSUB 1100
910 J=(J-64)+4:NN=NN-2:GOSUB 1130:RETURN
920 GOSUB 1100
930 J=J+68:NN=NN+2:GOSUB 1130:RETURN
940 REM PLAYER ADVANCE / DELETE
945 PRINT@(0+LEN(N$)+16),"  ";
950 J(I)=J:D(I)=D:I=I+1:GOSUB 1100
960 IF I>=4 THEN I=1:P=1
970 IF P=1 THEN J=J(I):D=D(I)
980 IF N$(I)="" THEN 950 ELSE GOTO 530
990 PRINT@(0+LEN(N$)+16),A;:RETURN
1000 PRINT"PRESS A KEY   "
1010 X$=INKEY$:IF X$="" THEN 1010
1020 RETURN
1100 REM CLEAR CHARACTER BEFORE MOVEMENT
1110 PRINT@J,CHR$(32);
1120 RETURN
1130 PRINT@J,CHR$(143),
1140 RETURN
1200 REM DROP A PLAYER ???
1210 IF I=1 THEN 1250
1220 IF I=2 THEN 1240
1230 IF J=J(I-2) OR J=J(I-1) THEN U=I:GOTO 1270
1240 IF J=J(I-1) OR J=J(I+1) THEN U=I:GOTO 1270
1250 IF J=J(I+1) OR J=J(I+2) THEN U=I:GOTO 1270
1260 GOTO 760
1270 IF J<>L OR K<>A THEN 1260
1280 REM PLAYER DELETED
1290 IF U=3 THEN 1330
1300 IF U=2 THEN 1340
1310 IF U=1 THEN 1350
1320 GOTO 1260
1330 IF J=J(I-2) THEN NX=I-2:GOTO 1360
1335 IF J=J(I-1) THEN NX=I-1:GOTO 1360
1340 IF J=J(I-1) THEN NX=I-1:GOTO 1360
1345 IF J=J(I+1) THEN NX=I+1:GOTO 1360
1350 IF J=J(I+1) THEN NX=I+1:GOTO 1360
```

```
1355 IF J=J(I+2) THEN NX=I+2:GOTO 1360
1360 PRINT@64,,
1370 PRINT"PLAYER ELIMINATED...";N$(NX);"      "
1380 N$(NX)="":FT=30:FF=3:GOSUB 1580
1390 GOTO 1500
1400 REM LOCATION SELECT
1410 IF R>=2 THEN 1430
1420 FOR W=1TO10:READ L(W):NEXT
1430 L=RND(10)
1440 IF L(L)=0 THEN 1430
1450 L=L(L):RETURN
1460 DATA 850,889,822,716,688
1470 DATA 621,594,472,420,411
1500 REM DELETE AND SELECT
1510 REM ANOTHER LOCATION
1520 L(R)=0:R=R+1
1530 IF R=3 THEN FF=10:FT=1:GOTO 1570
1540 GOSUB 1400
1550 J(NX)=0:FOR T=1TO1000:NEXT
1560 GOTO 940
1570 GOSUB 1580:GOTO 1640:REM WINNER
1580 A=150:FOR I=1TO3:A(I)=A
1590 A=A+50:NEXT:F=0
1600 FOR I=1TO3
1610 POKE 31265,30:POKE 31266,A(I)-FT:X=USR(0)
1620 NEXT
1630 IF F<>FF THEN F=F+1:GOTO 1600 ELSE RETURN
1640 PRINT@128,;
1650 PRINT"THE CHAMP IS....";
1660 IF N$(1)<>"" PRINT N$(1):GOTO 1690
1670 IF N$(2)<>"" PRINT N$(2):GOTO 1690
1680 IF N$(3)<>"" PRINT N$(3)
1690 FOR T=1TO1200:NEXT
1700 CLS
1710 INPUT"READY TO TRY ANOTHER TRIANGLE";Z$
1720 IF Z$<>"Y" AND Z$<>"YES" THEN 1750
1730 CLS:FOR I=1TO3:J(I)=0:NEXT:P=0
1740 RESTORE:GOTO 240
1750 PRINT
1760 PRINT"END OF TRIANGLE....":F=0
1770 POKE 31265,30:POKE 31266,120:X=USR(0)
1780 IF F<>15 THEN F=F+1:GOTO 1770
1790 END
```

Variables and Strings

N$(I) - Players' names
L(I),L - Selection of one location by the computer

W - Triangle graphics
J - Character position of current player
NN - For sound output (machine program)
A - User move number (value of X$)
K - Value of the distance from the start position to stop
D - For direction of character (around inside of triangle)
J(I) - For advance or elimination of a player
U - Player eliminated
F - Variable for sound output

Explanation of the Program Lines

Lines 10-230 display the general operating instructions.

Lines 240-280 request the three players' first names.

Lines 290-350 select the first player.

Lines 360-520 indicate who the first player is, display the triangle, and establish the sound routine.

Lines 530-620 keep track of the current player's location within the triangle.

Lines 630-690 accept the player's move.

Lines 700-930 move the player's character in the correct direction and keep it within the triangle.

Lines 940-1020 either advance or eliminate the player.

Lines 1100-1140 clear the character before movement.

Lines 1200-1390 eliminate a player if that player has landed on a computer selected location.

Lines 1400-1560 read the data for all the locations within triangle and then select one location.

Lines 1570-1790 print the winner's name and allows the users to try again.

MIND INVASION

The computer will try to invade your mind with as many moving graphics as possible. What you must do is move out of the way of the two blocks before they meet in the middle of the video and squash you. To gain any points at all, you must be constantly moving up or down. This motion is achieved by using the up and down arrows, of course. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
MIND INVASION

NUMBER OF PLAYERS? 5
PLAY OUTLINE NECESSARY? YES
```

EACH OF YOU 5 PLAYERS WILL
EXPERIENCE A NEW FORM OF MIND
INVASION. THE OBJECT OF THE GAME:
KEEP PLAYING AS LONG AS POSSIBLE
WITHOUT GETTING CRUSHED.
YOU, THE USER WILL BE LOCATED AT
THE CENTER OF THE VIDEO DISPLAY
IN THE FORM OF A '+' SIGN. YOUR
ONLY MOVEMENTS WILL BE STRAIGHT
UP OR STRAIGHT DOWN, ACCOMPLISHED
WITH THE UP ARROW AND DOWN
ARROW KEYS.

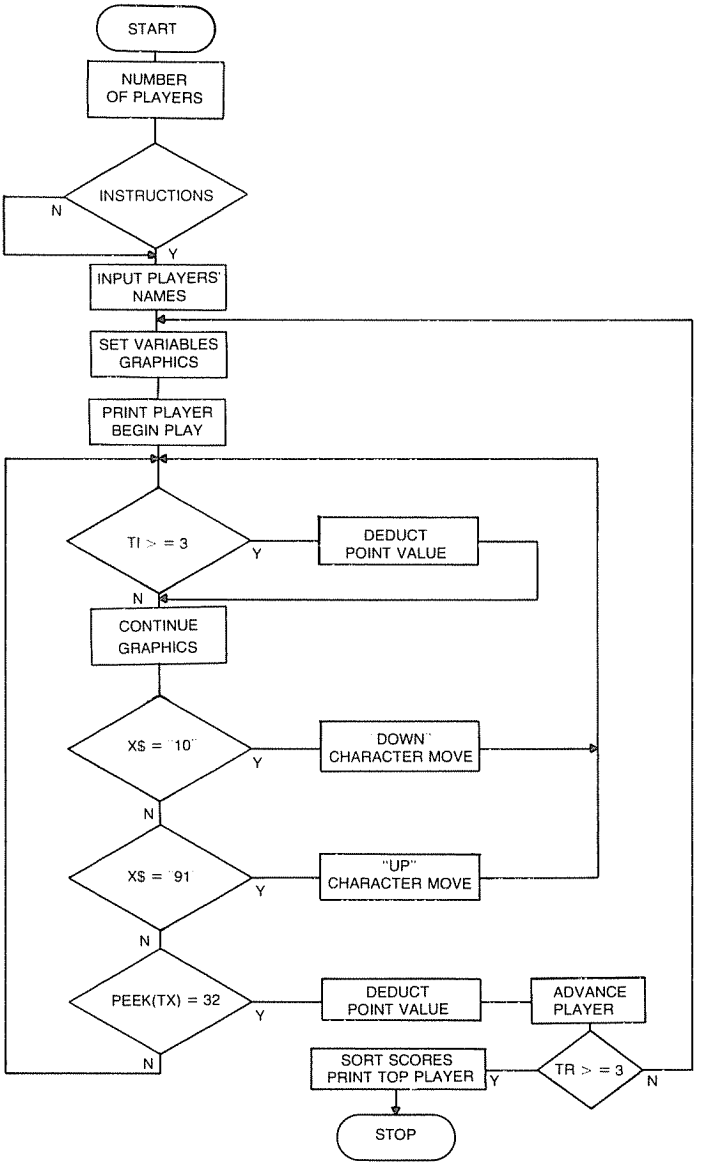
PRESS A KEY.....

TO TAKE ADVANTAGE OF ANY POINTS
YOU MUST BE CONSTANTLY MOVING
UP OR DOWN. THE OUTER BOUNDARIES
OF THE VIDEO WILL CONTAIN
FLASHING ASTERISKS FOR CONFUSION.
ALSO CONTAINED IN THOSE BOUNDARIES
ARE TWO BLOCKS THAT,
WITHOUT NOTICE, WILL MOVE VERY
QUICKLY TOWARD THE CENTER TO
YOUR LOCATION. IF YOU ARE SUCCESS-
FUL IN MOVING OUT OF THE WAY,
YOU CAN CONTINUE PLAYING. IF, ON
THE OTHER HAND, YOU GET SMASHED,
PLAY WILL ADVANCE TO THE NEXT
PLAYER. THIS IS ALSO TRUE IF
COLLIDE INTO THE BLOCKS.

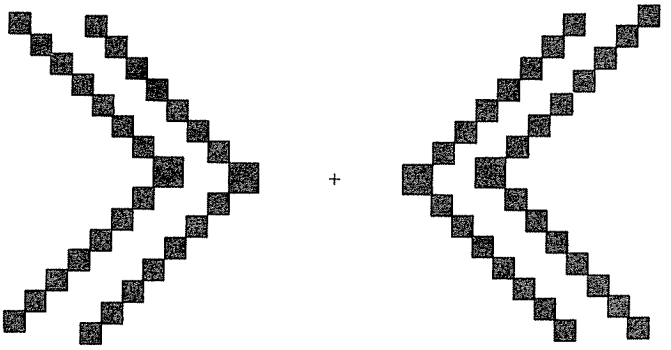
PRESS A KEY.....

THE GAME ENDS WHEN EACH
PLAYER HAS PLAYED A TOTAL OF 2 (TWO)
ROUNDS. THE PLAYER FINISHING WITH
THE HIGHEST SCORE WILL BE THE
WINNER.....

ENTER THE 5 PLAYERS
FIRST NAMES NOW



Flowchart for *Mind Invasion*.



Program Listing

```
 10 REM PROGRAM TITLE: MIND INVASION
 20 REM INSTRUCTION ROUTINE
 30 CLEAR 500:GOSUB 1500:J=1:TR=1
 40 REM OUTER GRAPHICS
 50 X=15360:Y=15423:K=191:PN=1:TX=Y+417:TL=TX
 60 POKE X,K:POKE Y,K:POKE X+10,K:POKE Y-10,K
 70 X=X+65:Y=(Y+64)-1:J=J+1
 80 IF J<>INT(K/22) THEN 60
 90 REM GOTO CHARACTER PRINT / IF DONE
100 IF J=16 THEN N=42:POKE TX,43:GOTO 140
110 POKE X,K:POKE Y,K:POKE X+10,K:POKE Y-10,K
115 IF PN=1 THEN A(1)=X:A(2)=Y:L=A(1):R=A(2)
120 X=(X+64)-1:Y=Y+65
130 J=J+1:PN=2:GOTO 100
140 PRINT@1000,"YOUR POINTS =";
150 PI=0
160 GOSUB 910
170 PRINT@960,"PLAYER. . .";N$(PZ);
200 REM CHARACTER GRAPHICS VARIABLES
210 J=7:E=448:F=505:T=1:E2=E:F2=F:TC=11:BC=907
220 KY=1:G=42:GOSUB 380:IF TI>=3 THEN 880
230 PRINT@E,W$;:PRINT@F,W$;:E1=E+64:F1=F+65
240 T=T+1:GOSUB 350
250 PRINT@E,L$;:PRINT@F,L$;
260 IF T>=2 THEN J=J-1:GOSUB 380:GOSUB 350
270 PRINT@E-64,W$;:PRINT@E1,W$;
280 PRINT@F-63,W$;:PRINT@F1,W$;
290 PRINT@E-64,L$;:PRINT@E1,L$;
300 PRINT@F-63,L$;:PRINT@F1,L$;:GOTO 600
310 E=E-64:F=F-63:E1=E2+128:F1=F2+130
```

```
320 E2=E2+64:F2=F2+65:T=T+1
330 REM RESET VARIABLES IF DONE
335 GOTO 700
340 IF T)=8 THEN 200 ELSE KY=KY+1:GOTO 260
350 REM FOR CHARACTER CLEAR
360 L$=STRING$(J,32):Q$=STRING$(G,42)
370 Q1$=STRING$(G,32):RETURN
380 W$=STRING$(J,N)
390 RETURN
400 X$=INKEY$:REM USER MOVEMENT
405 TI=TI+1:REM LIMIT COUNT WITHOUT MOVEMENT
410 IF X$="" THEN 340
420 TI=1:REM PLACEMENT / LIMIT / RESET
430 IF ASC(X$)=10 THEN 460
440 IF ASC(X$)=91 THEN 490
450 GOTO 340
460 POKE TX,32:TX=TX+64
470 IF ABS(TX-TL))=128 THEN TX=TX-64
480 POKE TX,43:U(KY)=ASC(X$):GOTO 340
490 POKE TX,32:TX=TX-64
500 IF ABS(TX-TL))=128 THEN TX=TX+64
510 POKE TX,43:U(KY)=ASC(X$):GOTO 340
600 REM TOP / BOTTOM CHARACTERS
610 PRINT@TC,Q$;
620 PRINT@BC,Q$;
630 PRINT@TC,Q1$;
640 PRINT@BC,Q1$;
650 TC=TC+65:BC=BC-63:G=G-2
660 REM REPLACE USER CHARCTER
670 IF PEEK(TX)=32 THEN POKE TX,43
680 GOTO 310
700 REM CRUSHER MOVEMENT
705 POKE A(1)+10,32:POKE A(2)-10,32
710 POKE L,32:POKE R,32
720 IF ABS(L-R)=1 THEN 750
730 L=L+8:R=R-8
740 POKE L,K:POKE R,K:GOTO 400
750 L=A(1):R=A(2):GOTO 800
760 POKE L,K:POKE L+10,K
770 POKE R,K:POKE R-10,K
780 REM RECYCLE
790 GOTO 400
800 REM CRUSHED ?
810 IF PEEK(TX)=32 THEN 840
820 IF TI<=2 AND U(KY)<>U(KY-1) THEN PI=PI+ABS
    (TX-TL):GOSUB 910
830 GOTO 760
840 REM YES.....CRUSHED
```

```
 850 PRINT@960,"(( C R U S H E D ))";
 860 POKE TX,32
 870 FOR LX=1TO2000:NEXT:GOTO 950
 880 REM LIMIT REACHED / NO MOVEMENT
 890 TI=0
 900 PI=PI-ABS(TL-TX):GOSUB 910:GOTO 920
 910 PRINT@1014,PI;:RETURN
 920 GOTO 230
 950 REM PLAYER ADVANCE / END
 960 TP(PZ)=PI:PI=0:PZ=PZ+1
 970 IF PZ>PL THEN GOSUB 1200:TR=TR+1:PZ=1
 980 IF TR)=3 THEN GOSUB 1240:GOTO 1010
 990 FOR LX=1TO14:PRINT:NEXT
1000 GOSUB 2100:CLS:J=1:GOTO 40
1010 REM END / HIGH POINT WINNER
1020 T=0:H=1:IF H=PL THEN 1120
1030 IF FT(H))=FT(H+1) THEN 1080
1040 FT=FT(H):FT(H)=FT(H+1)
1050 FT(H+1)=FT:N$=N$(H):N$(H)=N$(H+1)
1060 N$(H+1)=N$
1070 T=1
1080 H=H+1
1090 IF H(PL THEN 1030
1100 IF T=1 THEN 1020:REM NOT REFINED
1110 GOTO1300
1120 CLS
1130 PRINT
1140 PRINT"YOUR TOTAL POINTS, AFTER 2"
1150 PRINT"ROUNDS =";TP(H)+WE(H)
1160 GOTO 1390
1200 REM TALLY FIRST ROUND POINTS
1210 IF TR)=2 RETURN
1220 FOR I=1TOPL
1230 WE(I)=TP(I):NEXT:RETURN
1240 REM FINAL TALLY
1250 FOR I=1TOPL
1260 FT(I)=WE(I)+TP(I)
1270 NEXT
1280 RETURN
1300 REM TOP WINNER (MORE THAN 1 PLAYER)
1310 CLS
1320 I=1
1330 PRINT"THE TOP WINNER: ";N$(I);" WITH";
1340 PRINT FT(I);"POINTS."
1350 PRINT"FOLLOWED BY:"
1360 FOR I=2TOPL:PRINT N$(I);" WITH";
1370 PRINT FT(I);"POINTS."
1380 NEXT
```

```
1390 PRINT
1400 PRINT"END OF PROGRAM RUN."
1410 END
1500 REM INSTRUCTIONS
1510 I$="MIND INVASION"
1520 I=5:CLS
1530 FOR LX=1TOLEN(I$)
1540 PRINTTAB(I) MID$(I$,LX,1);
1550 FOR Y=1TO50:NEXT Y
1560 I=I+1:NEXT LX
1570 FOR Y=1TO1000:NEXT
1580 PRINT
1590 INPUT"NUMBER OF PLAYERS (1-10)";PL
1600 IF PL<=0 OR PL>=11 THEN 1590
1610 PRINT"PLAY OUTLINE NECESSARY";
1620 INPUT X$
1630 IF X$="N" OR X$="NO" THEN 2040
1640 PRINT
1650 PRINT"EACH OF YOU";PL;"PLAYERS WILL"
1660 PRINT"EXPERIENCE A NEW FORM OF MIND"
1670 PRINT"INVASION. THE OBJECT OF THE GAME:"
1680 PRINT"KEEP PLAYING AS LONG AS POSSIBLE"
1690 PRINT"WITHOUT GETTING CRUSHED."
1700 PRINT"YOU, THE USER WILL BE LOCATED AT"
1710 PRINT"THE CENTER OF THE VIDEO DISPLAY"
1720 PRINT"IN THE FORM OF A '+' SIGN. YOUR"
1730 PRINT"ONLY MOVEMENTS WILL BE STRAIGHT"
1740 PRINT"UP OR STRAIGHT DOWN, ACCOMPLISHED"
1750 PRINT"WITH THE UP ARROW AND DOWN"
1760 PRINT"ARROW KEYS."
1770 PRINT
1780 PRINT"PRESS A KEY..... "
1790 X$=INKEY$:IF X$="" THEN 1790
1800 PRINT
1810 PRINT"TO TAKE ADVANTAGE OF ANY POINTS"
1820 PRINT"YOU MUST BE CONSTANTLY MOVING,"
1830 PRINT"UP OR DOWN. THE OUTER BOUNDRIES"
1840 PRINT"OF THE VIDEO WILL CONTAIN"
1850 PRINT"FLASHING ASTERISKS, FOR CONFUSION."
1860 PRINT"ALSO CONTAINED IN THOSE BOUNDRIES"
1870 PRINT"ARE TWO BLOCKS THAT,"
1880 PRINT"WITHOUT NOTICE, WILL MOVE VERY"
1890 PRINT"QUICKLY TOWARD THE CENTER TO"
1900 PRINT"YOUR LOCATION. IF YOU ARE SUCCESS-"
1910 PRINT"FUL IN MOVING OUT OF THE WAY"
1920 PRINT"YOU CAN CONTINUE PLAYING. IF, ON"
1925 FOR LX=1TO2000:NEXT
1930 PRINT"THE OTHER HAND, YOU GET SMASHED"
```

```
1940 PRINT"PLAY WILL ADVANCE TO THE NEXT"
1950 PRINT"PLAYER. THIS IS ALSO TRUE IF"
1955 PRINT"YOU COLLIDE INTO THE BLOCKS."
1960 FOR LX=1TO2000:NEXT
1970 PRINT"PRESS A KEY....."
1980 X$=INKEY$:IF X$="" THEN 1980 ELSE PRINT
1990 PRINT"THE GAME ENDS WHEN EACH"
2000 PRINT"PLAYER HAS PLAYED A TOTAL OF TWO"
2010 PRINT"ROUNDS. THE PLAYER FINISHING WITH"
2020 PRINT"THE HIGHEST POINTS WILL BE THE"
2030 PRINT"WINNER....."
2040 PRINT
2050 PRINT"NOW ENTER THE";PL;"PLAYERS"
2060 PRINT"FIRST NAMES"
2070 FOR I=1TOPL:INPUT N$(I)
2080 NEXT
2090 PZ=1
2100 PRINT N$(PZ);" WILL NOW PLAY, PRESS"
2105 IF PZ>=2 OR TR>=2 THEN 2150
2110 PRINT"A KEY WHEN READY....."
2120 X$=INKEY$:IF X$="" THEN 2120
2130 CLS
2140 RETURN
2150 PRINT"THE ENTER KEY WHEN READY";
2160 INPUT X$:TI=0:RETURN
```

Variables and Strings

X - For the poke graphics for the left side
Y - For the poke graphics for the right side
K - Character code for X,Y and above
PN - Counter to set subscripted variables A(1) and A(2) for block start position
L,R - For left and right movement of blocks
TX,TL - For user character placement and used to signify character area. TX is also for the user character if it is crushed.
PI - Total points for each player
PZ - Counter for the correct player
E,F,T,E2,F2 - For the Print @ graphics for the outer areas of the display
TC,BC - For the top center and bottom center graphics
KY - For key closure counter. Subscripted variable U (U(KY)) will insure that before any points are given, the user isn't pressing the same direction key repeatedly
G - A constant used for the number of characters for upper/lower graphics

TI - Counter for point deduction, user asleep?
T - For outer, upper and lower Print @ graphics
LX - For time loop
TP - For total points received by current player
PZ - Counter for number of players
TR - Total rounds played
H - Counter for player
FT - For players total points after 2 rounds
WE - Points obtained during first round

Explanation of the Program Lines

Line 30 clears 500 bytes of memory for string storage and branches to the instructions.

Lines 40-80 poke the graphic blocks on the left and right sides of the video.

Line 100 insures that all poke graphics have been completed and then branches to the start of the game.

Lines 110-130 continues the poke graphics if they are done.

Lines 140-170 start the game and print the player's name and score.

Lines 200-340 (and lines 600-680) keep all the graphics moving.

Lines 350-390 establish strings made up of characters or blank spaces.

Lines 400-510 accept the user's up and down character movement.

Lines 700-790 move the crusher blocks (left and right) and reset them if they coincide.

Lines 800-870 checks whether or not the user is crushed. If he or she has been crushed, play advances to next person. If not, line 820 tests the current and preceding key closure and the variable TI (time limit). If each condition is met the user's score will increase by amount at line 950.

Lines 880-900 deduct an amount from users score if the time limit (TI) has been reached. This is to insure that the user's character doesn't remain in one location.

Lines 910-920 print the player's score and then recycle and continue play.

Lines 950-980 advance to the next player or terminate the game if all have played two rounds.

Lines 990-1000 are used for player advance.

Lines 1010-1100 place the scores (and players) in order from the highest score to the lowest score.

Line 1110 is used for the termination of the run.

Lines 1120-1160 display the score when only one person is playing.

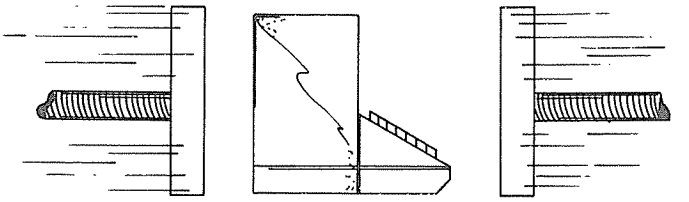
Lines 1200-1230 set the subscripted variable WE to the user's first round score.

Lines 1240-1280 set the subscripted variable FT to the user's final score.

Lines 1300-1410 print the name of the top winner of game and then rest of players' names in descending order.

Lines 1500-2160 are used for the number of players, their names and the general instructions.

Line 2105 tests the variables PZ and TR to determine which player is playing.

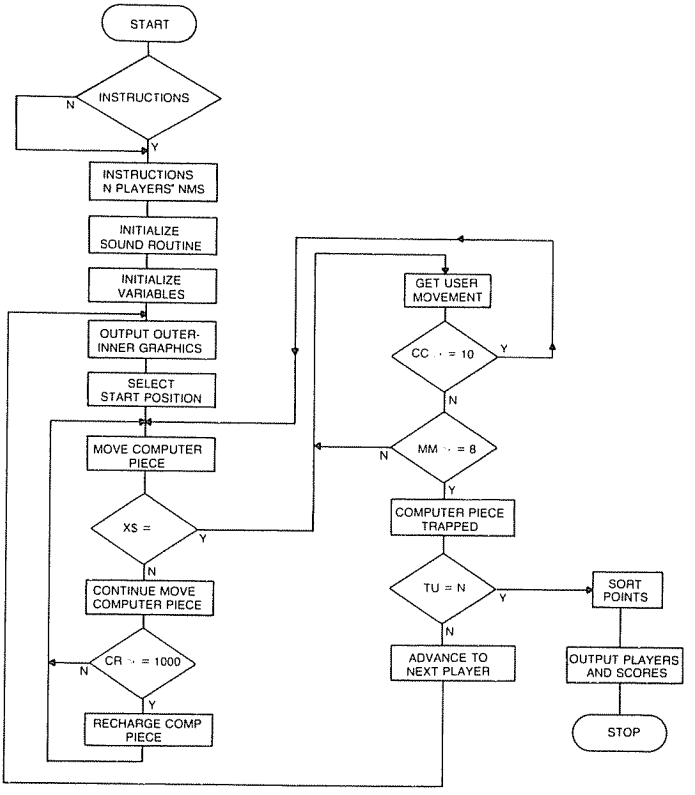


TRAPPED

You must trap the computer's piece. The drawbacks? The computer's piece can only be trapped in a left/right movement area. You must keep it pinned this way for a certain amount of time. Don't get burned by the computer's piece: six burns and you're out! There are, of course, other hidden obstacles, which you'll discover soon. Refer to pages 1-3 for information on the sound routine. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

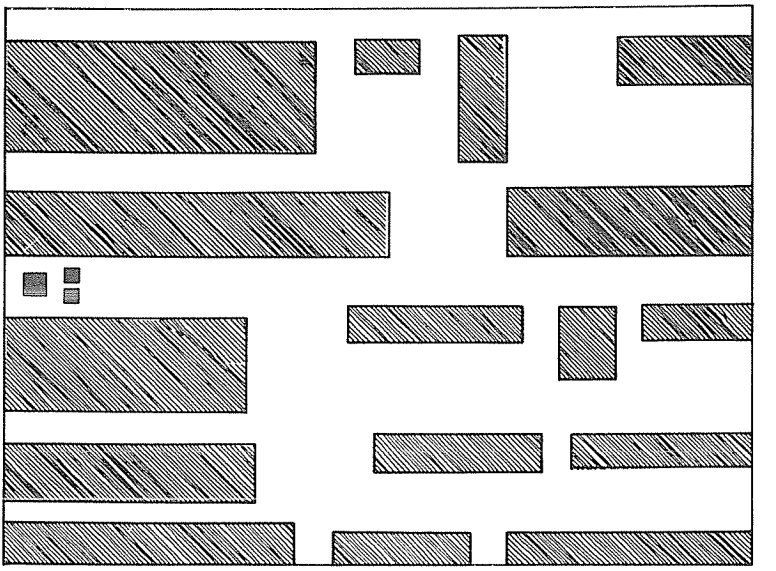
Sample Run

```
TRAPPED

LIST INSTRUCTIONS? YES

UP TO 10 PLAYERS CAN
TRY TO OUTWIT THE COMPUTER AND
MANEUVER ITS PIECE INTO A SITUATION
WHERE IT WILL BECOME TRAPPED.
THE COMPUTER'S PIECE WILL BE A
SOLID BLOCK, YOURS
WILL BE TWO SMALL BLOCKS,
CHR$(179). THE COMPUTER WILL MOVE
ITS PIECE ALL AROUND THE GAME
BOARD, CHANGING DIRECTION WHEN IT
SENSES AN OBSTACLE. YOU WILL
FOLLOW THE COMPUTER'S PIECE OR
YOU CAN MANEUVER IT TO A SECTION
OF THE GAME BOARD WHERE IT WILL BE

PRESS ENTER?

TRAPPED, MEANING IT CAN ONLY MOVE
```



Flowchart for *Trapped*.

LEFT AND RIGHT. THE COMPUTER'S PIECE WILL CONTAIN A 'CHARGE' IF YOU SIMULTANEOUSLY END AT THE SAME SPOT THAT THE COMPUTER DOES, YOU'LL GET BURNED. SIX BURNS AND YOU'RE OUT OF THE GAME. YOU WILL MOVE YOUR PIECE USING ALL FOUR ARROW KEYS. WHEN YOU MOVE THE COMPUTER WILL BLANK ITS PIECE. YOU WILL BE ALLOWED ONLY 10 CONSECUTIVE MOVES AT ONE TIME. ANYTIME YOU WISH TO MOVE YOUR PIECE, HOLD DOWN ANY ARROW KEY AND PRESS THE SPACE BAR.



PRESS ENTER?

POINTS WILL BE DETERMINED BY HOW
MANY MOVES IT TAKES YOU
TO TRAP THE COMPUTER'S PIECE.
REMEMBER, TO TRAP THE COMPUTER'S
PIECE, ONLY ALLOW IT TO MOVE
LEFT AND RIGHT.

PRESS ENTER TO START?

Program Listing

```
 )10 REM PROGRAM TITLE: ( TRAPPED )
 20 CLS:CLEAR 100:DIM AA(12),PI(12),I$(12)
 30 PRINT CHR$(183);" TRAPPED ";CHR$(187)
 40 PRINT:INPUT"LIST INSTRUCTIONS";W$
 50 IF W$="Y" OR W$="YES" THEN GOSUB 3500
 60 CLS:GOSUB 3300:TU=1
 70 QQ=1:MM=0:CR=0:RANDOM
 80 POKE 16526,0:POKE 16527,122
 90 S1=31265:S2=S1+1:CC=0:BR=0:TP=1E+04:TR=TP
100 REM BORDER GRAPHICS
110 X=15360:Y=X+63:C=191:U=0
```

```
120 FOR I=XTOY:POKE I,149
130 POKE I+960,149:NEXT
140 FOR I=X+64 TO X+INT(Y/19)+85 STEP 64
150 POKE I,149:POKE I+63,149:NEXT:GOSUB 3450
160 REM INNER GRAPHICS
170 IN=X+128
180 R=IN+RND(40):R1=R:L=RND(2)
190 FOR I=IN TO R:POKE I,C
195 IF L=1 AND U<=3 THEN POKE I+64,C
198 NEXT
200 R2=RND(7):R3=R2+RND(15):GOTO 500
210 FOR I=R1+R2 TO R1+R3
220 POKE I,C:NEXT
230 GOTO 560
240 U=U+1
250 IF U<>5 THEN 180
260 REM GRAPHICS FROM RIGHT SIDE
270 FOR J=1TO RND(10)
280 FOR I=0TO4
290 IF R(I)=0 THEN 310
300 POKE R(I),C:R(I)=R(I)-1
310 NEXT I,J
320 REM START POSITION
330 P=RND(16255)
340 IF P<=15425 THEN 330
350 IF PEEK(P)<>32 OR PEEK(P+2)<>32 THEN 330
355 IF (PEEK(P-64)+PEEK(P+64))=382 THEN 330
360 CP=P+2:POKE P,143:POKE CP,179
370 REM MOVEMENT OF COMPUTER PIECE
380 S3=210:S4=25
390 IF PEEK(P+64)=32 THEN A=1:GOTO 440
395 POKE S1,30:POKE S2,190:X=USR(0)
400 IF PEEK(P-64)=32 THEN A=2:GOTO 440
405 POKE S1,30:POKE S2,200:X=USR(0)
410 IF PEEK(P-2)=32 THEN A=3:GOTO 440
415 POKE S1,30:POKE S2,S3:X=USR(0)
420 IF PEEK(P+2)=32 THEN A=4:GOTO 440
425 POKE S1,30:POKE S2,S3:X=USR(0)
430 GOTO 810
440 POKE P,32:IF PEEK(CP)=32 THEN 1700
442 X$=INKEY$:IF X$=" " THEN 900
445 GOTO 1100
450 ON A GOTO 460,470,480,490
460 P=P+64:POKE P,143:MM=0:GOTO 710
470 P=P-64:POKE P,143:MM=0:GOTO 400
480 P=P-2:POKE P,143:GOTO 760
490 P=P+2:POKE P,143:GOTO 420
500 REM WITHIN LIMITS (BOARD GRAPHICS)
```

```
510 IF ABS((R1+R3)-IN))=60 THEN 200
520 REM GAP TOO SMALL
530 IF ABS((R1+R2)-R)(=4 THEN 200
540 IF R3(=9 THEN 200
550 GOTO 210
560 REM POKE MORE TO RIGHT
570 IF ABS((IN+63)-(R1+R3)))=10 THEN 600
580 X=X+C+1:IN=X+128:POKE IN,C
590 GOTO 240
600 R4=RND(8):IF R4(=3 THEN 600
610 R4=R4+R1+R3
620 R5=R4+RND(15)
630 IF ABS(R5-R4)(=2 THEN 620
640 IF ABS(R5-IN))=62 THEN 680
650 FOR I=R4 TO R5:POKE I,C
660 IF L=1 AND U()4 THEN POKE I+64,C
670 NEXT
680 REM VARIABLES (RIGHT SIDE GRAPHICS)
690 FF=IN+62:R(U)=FF:N(U)=ABS(FF-R5)
700 GOTO 580
710 REM DOWN / UP - SEARCH LEFT / RIGHT
720 IF PEEK(P-2)=32 THEN 410
730 IF PEEK(P+2)=32 THEN 420
740 IF A=1 THEN 390
750 GOTO 400
760 REM LEFT / RIGHT - SEARCH DOWN / UP
770 IF PEEK(P+64)=32 THEN 390
780 IF PEEK(P-64)=32 THEN 400
790 IF A=3 THEN 410
800 GOTO 420
810 REM TRAPPED (COUNTER)
815 A(QQ)=A
820 IF A(QQ)=A(QQ-1) THEN 880
830 IF MM)=8 THEN 1240
840 IF MM)=4 THEN S3=S3+10:CR=CR+10
850 IF S3)=255 THEN S3=255
860 QQ=QQ+1:IF QQ)=3 THEN QQ=1
870 GOTO 390
880 IF PEEK(P+2)=179 OR PEEK(P-2)=179 THEN MM=MM+1
890 GOTO 830
900 REM USER PIECE / MOVE PER KEY
910 KE=PEEK(15340):REM SCAN KEYBOARD
920 IF KE=136 THEN 970
930 IF KE=144 THEN 990
940 IF KE=192 THEN 1010
950 IF KE=160 THEN 1030
960 CC=0:GOTO 450
970 IF PEEK(CP-64)()32 THEN 960
```

```
980 POKE CP,32:CP=CP-64:GOTO 1050
990 IF PEEK(CP+64)()32 THEN 960
1000 POKE CP,32:CP=CP+64:GOTO 1050
1010 IF PEEK(CP+2)()32 THEN 960
1020 POKE CP,32:CP=CP+2:GOTO 1050
1030 IF PEEK(CP-2)()32 THEN 960
1040 POKE CP,32:CP=CP-2
1050 POKE CP,179:TP=TP-1
1060 POKE S1,30:POKE S2,195:X=USR(0)
1070 REM USER'S MOVES (COUNTER)
1080 CC=CC+1:IF CC)=10 THEN CC=0:GOTO 450
1090 GOTO 900
1100 REM SOUND / COMPUTER PIECE
1110 REM CHARGE RATE (DIMINISH BY PLUS)
1120 POKE S1,30:POKE S2,S4:X=USR(0)
1130 CR=CR+1
1140 IF CR)=1000 THEN GOSUB 1450:GOTO 1180
1150 IF CR)=990 THEN S4=S4+10
1160 IF S4)=255 THEN S4=255
1170 GOTO 450
1180 REM RECHARGE COMPUTER PIECE
1190 POKE P,143:FOR C=1 TO ((S4*2)-100)
1200 POKE S1,30:POKE S2,15:X=USR(0)
1210 NEXT:CR=0:S4=25:POKE P,32
1220 GOSUB 1510:GOSUB 3450:GOTO 450
1240 REM TRAPPED ?
1250 REM CHECK CHARGE RATE FIRST
1260 IF CR)=925 THEN 1280
1270 GOTO 1290
1280 CR=1000:GOTO 1370
1290 REM TRAPPED COMP. PIECE
1300 L=0
1310 POKE P,32:FOR TI=1TO100:NEXT
1320 POKE P,143:FOR TI=1TO50
1330 POKE S1,30:POKE S2,20:X=USR(0)
1340 NEXT
1350 L=L+1:IF L(=2 THEN 1310
1360 GOTO 1600
1370 GOSUB 1450:POKE P,32:MM=0
1380 P=P+6
1390 GOTO 450
1450 REM MESSAGE
1460 MG=16345:M$="RECHARGING"
1470 FOR I=1TO LEN(M$)
1480 POKE MG,ASC(MID$(M$,I,1))
1490 MG=MG+1:NEXT
1500 POKE MG,32:RETURN
1510 MG=16345
```

```
1520 FOR I=1TO LEN(M$)
1530 POKE MG,149
1540 MG=MG+1:NEXT
1550 POKE MG,149:RETURN
1600 REM MESSAGE (TRAPPED)
1610 MG=16345
1620 M$="** TRAPPED **"
1630 GOSUB 1470
1640 GOTO 1910
1700 REM USER "BURNED"
1710 MG=16345
1720 M$="* USER BURNED *"
1730 GOSUB 1470
1740 REM LEAVE USER POSITION AS-IS
1750 L=0:MM=0
1760 POKE CP,32:FOR TI=1TO100:NEXT
1770 POKE CP,179:FOR TI=1TO50
1780 POKE S1,30:POKE S2,18:X=USR(0)
1790 NEXT:IF L<=1 THEN L=L+1:GOTO 1760
1800 BR=BR+1:GOSUB 1510:GOSUB 3450
1810 IF BR>=6 THEN GOSUB 1830:GOTO 1870
1820 GOTO 445
1830 L=0
1840 FOR I=1TO45
1850 POKE S1,30:POKE S2,I+10
1860 X=USR(0):NEXT:CLS:RETURN
1870 PRINT
1880 PRINT"YOU'VE BEEN BURNED";BR;"TIMES"
1890 PRINT"SUCKER !!!":TP=0
1900 PRINT"YOUR OUT OF THE GAME...":GOTO 2000
1910 REM TRAPPED
1920 GOSUB 1830
1930 PRINT"YA GOT ME TRAPPED, TURKEY !!!"
1940 PRINT"YOUR TOTAL POINTS NOW EQUAL ";
1950 PRINT TP
1960 PRINT"YOU MADE A TOTAL OF";TR-TP;"MOVES"
1970 PRINT"BEFORE TRAPPING ME....."
1980 POKE S1,30:POKE S2,255
1990 X=USR(0)
2000 REM PLAYER ADVANCE / POINTS
2010 PI(TU)=TP
2020 IF TU=N THEN 2100
2030 TU=TU+1
2040 PRINT"KEY ENTER ";I$(TU);
2050 INPUT W$:CLS
2060 GOTO 70
2100 REM ALL PLAYED
2110 REM SORT AMTS / PLAYERS
```

```
2120 IF N=1 THEN 2220
2130 F=0:AA=1
2140 IF PI(AA)>=PI(AA+1) THEN 2190
2150 P=PI(AA):PI(AA)=PI(AA+1)
2160 PI(AA+1)=P:I$=I$(AA)
2170 I$(AA)=I$(AA+1):I$(AA+1)=I$
2180 F=1
2190 IF AA<N THEN AA=AA+1:GOTO 2140
2200 IF F=1 THEN 2130
2210 GOTO 2260
2220 PRINT"YOU'VE PLAYED BY YOURSELF, AGAINST"
2230 PRINT"THE COMPUTER. YOU WERE BURNED";BR
2240 PRINT"TIMES. FINAL TRAPPING SCORE:";PI(TU)
2250 GOTO 2330
2260 I=1
2270 PRINT"BEST TRAPPER.....";I$(I)
2280 PRINT"TOTAL POINTS =";PI(I)
2290 PRINT"FOLLOWING ";I$(I)
2300 FOR I=2TON:PRINT I$(I) ;
2310 PRINT" WITH A SCORE OF: ";
2320 PRINT PI(I)
2325 FOR TI=1TO1000:NEXT TI,I
2330 PRINT
2340 PRINT"END OF PROGRAM RUN....."
2350 FOR I=1TO100:POKE S1,30:POKE S2,I+20
2360 X=USR(0):NEXT:PRINT
2370 PRINT"UNLESS ALL WANT TO PLAY AGAIN ???"
2380 PRINT"IF SO, TYPE RUN."
2390 PRINT:FOR TI=1TO1500:NEXT
2400 PRINT"END
2410 END
3300 REM PLAYERS (AMTS. / NAMES)
3310 PRINT"ENTER NUMBER OF PLAYERS"
3320 INPUT N
3330 IF N<=0 OR N>=11 THEN 3310
3340 PRINT"NOW ENTER THE";N;"PLAYER'S"
3350 PRINT"FIRST NAMES"
3360 FOR I=1TON
3370 INPUT I$(I):NEXT
3380 PRINT
3390 PRINT"CURRENT PLAYER'S NAME"
3400 PRINT"WILL BE PRINTED AT BOTTOM OF"
3410 PRINT"GAME BOARD....."
3420 FOR TI=1TO1200:NEXT
3430 CLS
3440 RETURN
3450 REM PLAYER
3460 MG=16348
```

```
3470 FOR I=1TO LEN(I$(TU))
3480 POKE MG,ASC(MID$(I$(TU),I,1)):MG=MG+1
3490 NEXT:POKE MG,32:RETURN
3500 REM INSTRUCTIONS
3510 PRINT" UP TO 10 PLAYERS CAN"
3520 PRINT"TRY TO OUTWIT THE COMPUTER AND"
3530 PRINT"MANEUVER ITS PIECE INTO A SITUATION"
3540 PRINT"WHERE IT WILL BECOME TRAPPED."
3550 PRINT"THE COMPUTER'S PIECE WILL BE A"
3560 PRINT"SOLID BLOCK, YOURS"
3570 PRINT"WILL BE TWO SMALL BLOCKS.";CHR$(179)
3580 PRINT"THE COMPUTER WILL MOVE"
3590 PRINT"ITS PIECE ALL AROUND THE GAME"
3600 PRINT"BOARD, CHANGING DIRECTION WHEN IT"
3610 PRINT"SENSES AN OBSTACLE. YOU WILL"
3620 PRINT"FOLLOW THE COMPUTER'S PIECE OR"
3630 PRINT"YOU CAN MANEUVER IT TO A SECTION"
3640 PRINT"OF THE GAME BOARD WHERE IT WILL BE"
3650 INPUT"PRESS ENTER";W$:CLS
3660 PRINT"TRAPPED, MEANING IT CAN ONLY MOVE"
3670 PRINT"LEFT AND RIGHT. THE COMPUTER'S PIECE"
3680 PRINT"WILL CONTAIN A 'CHARGE'. IF YOU"
3690 PRINT"SIMULTANEOUSLY END AT THE SAME SPOT"
3700 PRINT"THAT THE COMPUTER DOES, YOU'LL"
3710 PRINT"GET 'BURNED'. SIX BURNS AND YOU'RE"
3720 PRINT"OUT OF THE GAME. YOU WILL MOVE"
3730 PRINT"YOUR PIECE USING ALL FOUR ARROW"
3740 PRINT"KEYS. WHEN YOU MOVE THE COMPUTER"
3750 PRINT"WILL 'BLANK' IT'S PIECE. YOU"
3760 PRINT"WILL BE ALLOWED ONLY 10 CON-"
3770 PRINT"SECUTIVE MOVES AT ONE TIME."
3780 PRINT"ANYTIME YOU WISH TO MOVE YOUR"
3790 PRINT"PIECE, HOLD DOWN ANY ARROW KEY"
3800 PRINT"AND PRESS THE SPACE BAR"
3810 INPUT"PRESS ENTER";W$:CLS
3820 PRINT"POINTS WILL BE DETERMINED BY HOW"
3830 PRINT"MANY MOVES IT TAKES THE USER"
3840 PRINT"TO TRAP THE COMPUTER'S PIECE."
3850 PRINT"REMEMBER, TO TRAP THE COMPUTER'S"
3860 PRINT"PIECE, ONLY ALLOW IT TO MOVE"
3870 PRINT"LEFT AND RIGHT."
3880 INPUT"KEY ENTER TO START";W$
3890 CLS.RETURN
```

Variables and Strings

TU - Used in conjunction with user's name to identify player
QQ - Used for left/right movement of computer piece

MM - Number of times computer piece moves left and right before final entrapment by the user; used in conjunction with A(QQ)
CR - Charge rate of computer's piece
S1,S2 - For sound location, machine language routine
CC - User's number of moves per key closure
BR - Number of burns player receives
TP - Total number of points allowable: one deducted for each user move
TR - Used in final message for number of moves player made before trapping computer piece
X,Y,C,U - Used for graphics and the game board
IN,R,R1,R2,R3,R4,R5,L,FF,R(U),N(U) - Determine the length and width of the inner poke graphics (left to right, right to left)
P - Used to determine the starting location of the computer piece
CP - Users start location, two more than the variable P
S3,S4 - Used in conjunction with computer piece for extra sound, along with variables S1 and S2
A — Determines route that computer's piece will take in conjunction with P (peek) arguments
KE - Key closure for user's moves. The arrow key must be pressed in conjunction with space bar for correct entry
L - Used again for the length of sound output
MG - Poke location for print messages
PI(TU) - For each players point amount, will return 0 (zero) if BR>=6 (user burned 6 times)
I$(TU) - Player's name, turn
N - Number of players (total)

Explanation of the Program Lines

Line 50 checks the entry of W$ for the print of the instructions at lines 3500-3890. Lines 3300-3440 (a GOSUB) will allow the entry of up to 10 players and their names.

Line 70 initializes some of the variables for each player.

Line 80 initializes the machine language subroutine.

Lines 90-150 poke the outer graphics and then branch to the subroutine in lines 3450-3490 for the printing of the current player's name within the graphics.

Lines 160-310 establish the inner graphics. When the variable L=1, a larger (double CHR$(191)) will be poked. Lines 500-550 work in association with these lines to keep the graphics within limits. Also used with the inner graphics are lines 560-670, which move the graphics more to the right. Finally, the game board

graphics use lines 680-700 to establish the variables for the inner graphics.

Lines 320-360 select the computer's start location randomly and establishes the user's start location at two more than the computer's. The arguments contained here assure that the start location will be in a blank area and that no borders are above or below the start location.

Lines 370-490 are used for the movement of computer's piece and the sound output, which uses locations S1 and S2. These lines use a search-ahead routine making sure the area ahead will be blank. Also used in conjunction with these arguments are lines 710-800, which assure that the computer's piece will have an alternate route to take unless trapped. Lines 460 and 470, where MM = 0 (reset counter), insure that the computer will only be trapped in a left and right direction; i.e., it can only move left or right.

Line 442 tests for the user's key closure. Holding down any arrow key will branch the flow to line 900. If none of the arguments are met (holding down the space bar, along with the selected arrow key) the program branches back to the computer piece movement. If any of the conditions are satisfied, the program makes sure that movement is toward a blank space, or else the movement cannot occur.

Line 1050, where TP=TP−1, subtracts one point for each move from the total of 10,000 points that the player is allowed.

Line 1060 outputs the user's sound.

Lines 1070-1090 count the number of moves the player makes. These lines will only allow 10 consecutive moves at one time. When this counter reaches a count of 10, the program resumes with the computers movements in line 450.

Lines 800-890 test to see if the computer's piece is actually being trapped (line 880). If it is, the variable MM is increased by one. If the count reaches 4, the pitch of the computer's sound will become higher and its charge will also increase. If the total count reaches 8 the computer's piece is trapped, and program flow branches to line 1240.

Lines 1100-1170 increase and test the computer's charge. If the charge rate is greater than or equal to 990, then the pitch will change for the computer's piece. When the charge rate reaches 1000, play comes to a halt until piece is recharged at lines 1180-1220.

Lines 1240-1270 also test the charge rate of the computer's piece. If it is greater than or equal to 925, the charge rate will

automatically become 1000. This will give the computer a break by advancing its piece 6 places to the right in lines 1280 to 1370.

Lines 1290-1360 are used for the final trapping of the computer's piece. Play will then branch to line 1600.

Lines 1420-1820 are for the print messages at the lower part of the game board. They also test burn amounts (BR) that the user has received. If BR is greater than or equal to 6, program flow branches to line 1870, where the player loses all his points, and the next player is given a chance to play (if applicable).

Lines 1910-1990 print the messages if the computer's piece is trapped.

Lines 2000-2060 advance to the next player, if the condition TU= N is not met. Scores points are also set with with subscripted variable PI(TU).

Lines 2100-2200 sort all players according to their scores, placing the highest scorer first and so on.

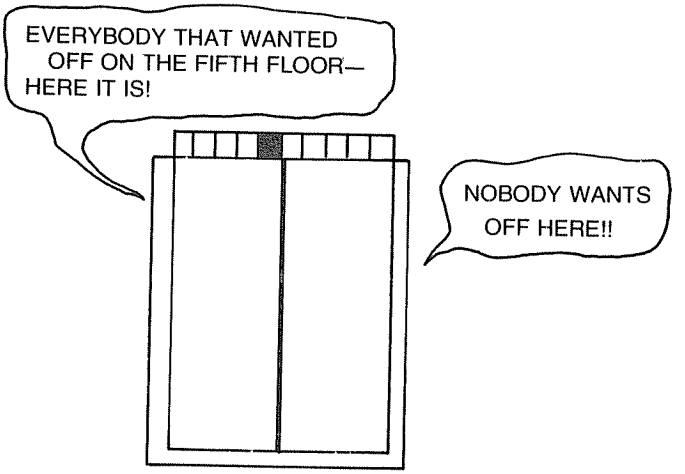
Lines 2220-2250 displays a message when there is only one player.

Lines 2260-2410 print the players' names and their scores.

Lines 3300-3440 allow the entry of up to 10 players and their first names.

Lines 3450-3490 print the player's first name at the lower part of the game board when it is his or her turn.

Lines 3500-3890 print the general instructions.



ELEVATOR

Guess what? You've been fired from your present job! Guess what else? The computer has hired you to be an elevator attendant! If you don't have a sharp memory, you'll get paid nothing. You see, you must remember what is on every floor and who wants off there. Some of the customers might even give you some tips! All is not lost, not yet that is... Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
    *** ELEVATOR ***

INSTRUCTIONS REQUIRED (Y/N)? Y

FIRST NAME? FUZZY

PRESENT OCCUPATION? BELLHOP
YOU HAVE JUST BEEN FIRED AS A BELLHOP!!
BUT, YOU ARE IN LUCK !! THE
COMPUTER HAS JUST HIRED YOU TO
BE AN ELEVATOR ATTENDANT. YOUR
JOB WILL BE QUITE SIMPLE, FUZZY.
ALL YOU HAVE TO DO IS REMEMBER
ALL THE FLOORS (16 TOTAL)
```

AND WHAT IS CONTAINED ON EACH
OF THEM, SO YOU CAN TAKE ALL
OF MY CUSTOMERS TO THEIR
RIGHT LOCATION.....
PRESS A KEY...

I ALMOST FORGOT TO TELL YOU, ALL
CUSTOMERS, WHEN STEPPING ONTO THE
ELEVATOR, WILL TELL YOU WHAT THEY
WANT; THEY WILL ONLY TELL YOU THIS
ONCE. FOR EVERY CUSTOMER YOU
PLACE ON THE RIGHT FLOOR, YOU'LL
BE PAID $10.00, THAT IS THE ONLY
WAY YOU'LL GET PAID. CUSTOMERS
WILL BE GETTING ON THE ELEVATOR
AT DIFFERENT STOPS; YOU MUST
REMEMBER THESE ALSO, FUZZY.
PRESS A KEY...
HERE ARE THE FLOORS AND TITLES:
2) WOMEN'S CLOTHES
3) MEN'S CLOTHES
4) CHILDREN'S CLOTHES
5) INFANT'S CLOTHES
6) TOYS
7) CARPETS

8) COMPUTERS

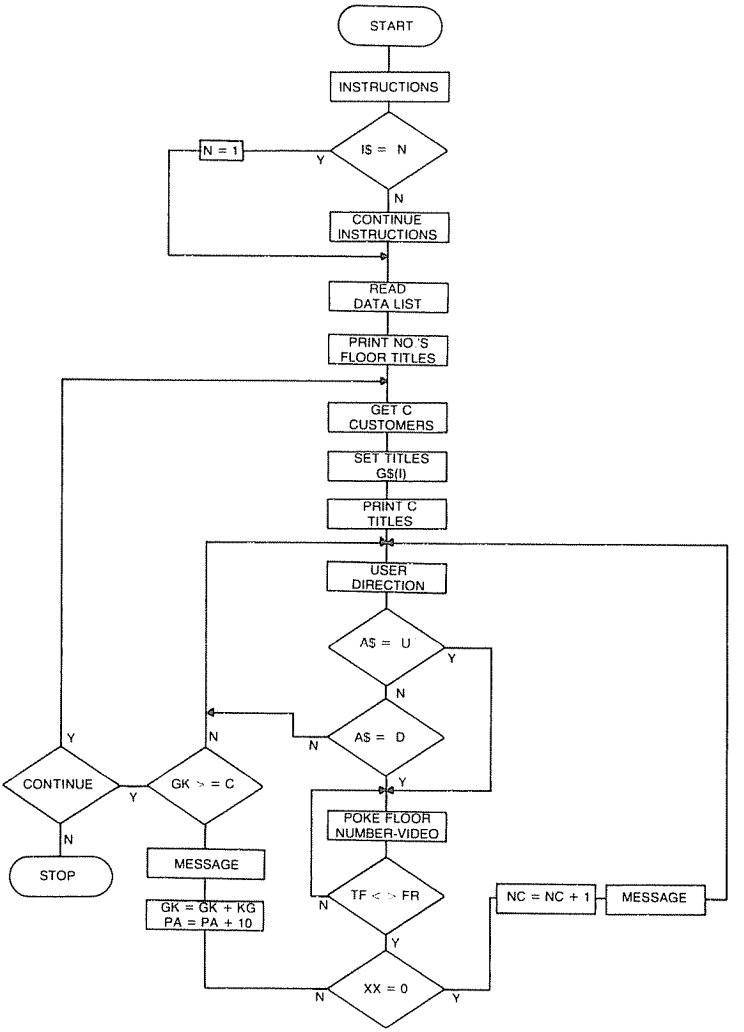
9) APPLIANCES

10) FURNITURE
11) HARDWARE SUPPLIES
12) HOBBY SUPPLIES
13) AUTO SUPPLIES
14) OFFICE SUPPLIES
15) BATHROOM ITEMS
16) BEDROOM ITEMS
YOU SHOULD BE ALL SET, FUZZY
FUZZY, YOU NOW HAVE 7 CUSTOMER(S)
GETTING ON MY ELEVATOR.

THE CUSTOMERS' DESTINATION(S):

COMPUTER FLOOR
AUTO SUPPLIES
CARPET FLOOR
HARDWARE SUPPLIES
HARDWARE SUPPLIES
COMPUTER FLOOR
INFANT'S CLOTHES
PRESS A KEY...
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 5
1 OF THE CUSTOMER(S) JUST
STEPPED OFF OF MY ELEVATOR, FUZZY.
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 7
1 OF THE CUSTOMER(S) JUST
STEPPED OFF OF MY ELEVATOR, FUZZY.
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: DOWN
FLOOR NUMBER 5
FUZZY NO CUSTOMERS WANTED OFF
ON THIS FLOOR...
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 17
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 10
FUZZY NO CUSTOMERS WANTED OFF
ON THIS FLOOR...
PRESS U FOR <UP>

PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: DOWN
FLOOR NUMBER 9
FUZZY NO CUSTOMERS WANTED OFF
ON THIS FLOOR...
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: DOWN
FLOOR NUMBER 9
YOU ARE ON THAT FLOOR, FUZZY!!
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: DOWN
FLOOR NUMBER 8
2 OF THE CUSTOMER(S) JUST
STEPPED OFF OF MY ELEVATOR, FUZZY.
FUZZY YOU JUST HAD 5 MORE
CUSTOMERS STEP ONTO THE
ELEVATOR. THE DESTINATION(S):
WOMEN'S CLOTHES
CARPET FLOOR
BATHROOM ITEMS
BEDROOM ITEMS
HARDWARE SUPPLIES
PRESS A KEY...
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 11
3 OF THE CUSTOMER(S) JUST
STEPPED OFF OF MY ELEVATOR, FUZZY.
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 7
IF YOU WANT TO GO UP, USE A
HIGHER NUMBER THAN THIS FLOOR.



Flowchart for *Elevator*.

```
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 12
FUZZY, NO CUSTOMERS WANTED OFF
ON THIS FLOOR...
FUZZY, YOU JUST HAD 4 MORE
```

```
CUSTOMERS STEP ONTO THE
ELEVATOR. THE DESTINATION(S):
MEN'S CLOTHES
FURNITURE FLOOR
WOMEN'S CLOTHES
GROUND FLOOR
PRESS A KEY...
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.
DIRECTION: UP
FLOOR NUMBER 13
1 OF THE CUSTOMER(S) JUST
STEPPED OFF OF MY ELEVATOR, FUZZY.
FUZZY, YOU JUST HAD 4 MORE
CUSTOMERS STEP ONTO THE
ELEVATOR. THE DESTINATION(S):
WOMEN'S CLOTHES
WOMEN'S CLOTHES
TOY FLOOR
MEN'S CLOTHES
PRESS A KEY...
PRESS U FOR <UP>
PRESS D FOR <DOWN>
THEN ENTER THE FLOOR NUMBER.

STOP
```

This is not a complete run.

Program Listing

```
 10 REM PROGRAM TITLE: ELEVATOR
 20 CLS:W$="$$######.##"
 30 DIM F(60),F$(60),G$(60)
 40 RANDOM
 50 PRINT TAB(10)"*** ELEVATOR ***"
 60 PRINT
 70 PRINT"INSTRUCTIONS REQUIRED (Y/N)";
 80 INPUT I$:IF I$="N" THEN N=1
 90 INPUT"FIRST NAME";N$
100 PRINT:IF N=1 THEN 230
110 INPUT"PRESENT OCCUPATION";P$
120 PRINT"YOU HAVE JUST BEEN FIRED AS A ";
```

```
125 PRINT P$;"!!"
130 PRINT"BUT, YOU ARE IN LUCK !! THE"
140 PRINT"COMPUTER HAS JUST HIRED YOU TO"
150 PRINT"BE AN ELEVATOR ATTENDANT. YOUR"
160 PRINT"JOB WILL BE QUITE SIMPLE, ";N$
170 PRINT"ALL YOU HAVE TO DO IS REMEMBER"
180 PRINT"ALL THE FLOORS (16 TOTAL)"
190 PRINT"AND WHAT IS CONTAINED ON EACH"
200 PRINT"OF THEM, SO YOU CAN TAKE ALL"
210 PRINT"OF MY CUSTOMERS TO THEIR"
220 PRINT"RIGHT LOCATION....."
230 REM WHAT'S ON EACH FLOOR
240 FOR I=2 TO 19:READ F$(I):NEXT
250 IF N=1 THEN 370 ELSE GOSUB 2150:PRINT
260 PRINT"I ALMOST FORGOT TO TELL YOU, ALL"
270 PRINT"CUSTOMERS, WHEN STEPPING ONTO THE"
280 PRINT"ELEVATOR, WILL TELL YOU WHAT THEY"
290 PRINT"WANT, THEY WILL ONLY TELL YOU THIS"
300 PRINT"ONCE. FOR EVERY CUSTOMER YOU"
310 PRINT"PLACE ON THEIR RIGHT FLOOR, YOU'LL"
320 PRINT"BE PAID $10.00, THAT IS THE ONLY"
330 PRINT"WAY YOU'LL GET PAID. CUSTOMERS"
340 PRINT"WILL BE GETTING ON THE ELEVATOR"
350 PRINT"AT DIFFERENT STOPS; YOU MUST"
355 PRINT"REMEMBER THESE ALSO, ";N$;"."
360 GOSUB 2150
370 PRINT"HERE ARE THE FLOORS AND TITLES:"
380 FOR I=2 TO 16
390 PRINT I;") ";F$(I);
395 IF I=8 THEN TT=I*200:PRINT:GOSUB 2190
400 IF I)=2 AND I(=5 THEN 440
410 IF I)=6 AND I(=10 THEN PRINT:GOTO 470
420 IF I)=11 AND I(=14 THEN 450
430 IF I)=15 AND I(=16 THEN 460
440 PRINT" ";F$(17):GOTO 470
450 PRINT" ";F$(18):GOTO 470
460 PRINT" ";F$(19)
470 IF RX=1 THEN RETURN ELSE NEXT
475 REM VARIABLE RX FOR PRINT ELEMENTS
480 FOR I=1 TO 6:PRINT:NEXT:TT=TT+900
490 GOSUB 2190:CLS:C=0:GK=0
500 PRINT"YOU SHOULD BE ALL SET ";N$
510 TT=TT-900:GOSUB 2190:RX=1
520 REM GET CUSTOMERS
530 C=RND(PEEK(33)+1)
540 PRINT N$;" YOU NOW HAVE";C;"CUSTOMER(S)"
550 PRINT"GETTING ON MY ELEVATOR."
560 REM GET C RESPONSES
```

```
570 I=1
580 REM FLOOR NUMBER
590 F=RND(PEEK(49)+1):IF F=1 THEN 590
595 IF F=TF THEN 590
600 F(I)=F:I=I+1
610 IF I<=C THEN 590
620 REM FLOOR TITLES
630 FOR I=1 TO C
640 G$(I)=F$(F(I))
650 NEXT
660 GOSUB 2190:PRINT
670 PRINT"THE CUSTOMERS DESTINATION(S):"
680 PRINT:FOR I=1 TO C
690 PRINT G$(I);"";
700 F=F(I)
710 IF F>=2 AND F<=5 THEN 760
720 IF F>=11 AND F<=14 THEN 770
730 IF F>=15 AND F<=16 THEN 780
740 PRINT" FLOOR"
750 NEXT:GOTO 790
760 GOSUB 440:GOTO 750
770 GOSUB 450:GOTO 750
780 GOSUB 460:GOTO 750
790 GOSUB 2150
800 CLS
810 PRINT"PRESS U FOR (UP)"
820 PRINT"PRESS D FOR (DOWN)"
830 PRINT"THEN ENTER THE FLOOR NUMBER"
840 NN=0:REM DIRECTION
850 GOSUB 2160
860 D$=A$:IF D$="U" D$="UP":GOTO 870
865 IF D$="D" THEN D$="DOWN":GOTO 870
868 GOTO 950
870 PRINT"DIRECTION: ";D$
880 PRINT"FLOOR NUMBER";
890 INPUT FR
895 IF FR=1 AND TF=0 THEN 1280
900 IF FR<1 OR FR>16 THEN 800
905 IF FR=TF THEN 1280
910 IF FR<TF AND D$="UP" THEN 1260
915 IF FR>TF AND D$="DOWN" THEN V=1:GOTO 1260
920 GOSUB 2190:CLS:REM PAUSE BEFORE START
930 IF D$="UP" THEN 970
940 IF D$="DOWN" THEN 1040
950 PRINT"SELECT A DIRECTION ";N$;"!!"
960 PRINT:GOTO 810
970 REM "UP"
980 IF TF>=17 THEN 1030
```

```
 990 GOTO 1300
1000 PRINT"YOU ARE ON THE ";T$;" FLOOR ";N$
1010 PRINT"YOU CAN NOT GO ANY ";H$;"!!"
1020 GOTO 960
1030 T$="TOP":H$="HIGHER":GOTO 1000
1040 IF TF<=1 THEN 1060
1050 GOTO 1070
1060 T$="BOTTOM":H$="LOWER":GOTO 1000
1070 REM "DOWN" CONTINUED (MOVEMENT)
1080 FS=P-512:NN=1
1085 IF FF)=58 AND FF(=64 THEN 1220
1090 POKE P,FF
1100 FOR T=1 TO 40:NEXT
1110 POKE P,32:POKE P+1,32:P=P-64
1120 IF P<>FS THEN 1085
1130 REM CONTINUE IF NOT ON FLOOR
1140 IF TF<>FR THEN 1160
1150 GOTO 1500
1160 FS=FS-448
1170 IF P<=15390 THEN 1190
1180 GOTO 1085
1190 POKE P,32:POKE P+1,32:P=16350
1200 FS=P-512:FF=FF-1:TF=TF-1
1210 GOTO 1085
1220 REM FLOORS ABOVE 9
1230 POKE P,49:POKE P+1,FF-10
1240 IF NN=1 THEN 1100
1250 GOTO 1340
1260 PRINT"IF WANTING TO GO ";D$;", USE A"
1265 IF V=1 V=0:PRINT"USE A LOWER NUMBER.":GOTO 1275
1270 PRINT"HIGHER NUMBER THAN THIS FLOOR."
1275 GOTO 1290
1280 PRINT"YOU ARE ON THAT FLOOR, ";N$;"!!"
1290 GOSUB 2190:GOTO 800
1300 REM "UP" MOVEMENT
1310 IF TF=0 THEN 1460
1320 FS=P+512:IF FS)=16350 THEN FS=16350
1325 IF FF)=58 AND FF(=64 THEN 1220
1330 POKE P,FF
1340 FOR T=1 TO 50:NEXT
1350 POKE P,32:POKE P+1,32:P=P+64
1360 IF P<>FS THEN 1325
1370 REM CONTINUE IF NOT ON FLOOR
1380 IF TF<>FR THEN 1400
1390 GOTO 1500
1400 FS=FS+448
1410 IF P)=16350 THEN 1430
1420 GOTO 1325
```

```
1430 POKE P,32:POKE P+1,32:P=15390
1440 FS=P+512:FF=FF+1:TF=TF+1
1450 GOTO 1325
1460 TF=1:P=15902:FF=49
1470 FS=P+448:GOTO 1325
1500 REM ON FLOOR REQUESTED
1510 IF FF)=58 AND FF(=64 THEN 1530
1520 POKE P,FF:GOTO 1540
1530 POKE P,49:POKE P+1,FF-10
1540 FOR I=1 TO C
1550 IF FR=F(I) THEN 1570
1560 NEXT:GOTO 1600
1570 PA=PA+10:KG=KG+1
1580 F(I)=0:XX=1
1590 GOTO 1560
1600 REM DID CUSTOMERS WANT OFF HERE
1610 IF XX=0 THEN 1630
1620 XX=0:GOTO 1770
1630 PRINT N$;" NO CUSTOMERS WANTED OFF"
1640 PRINT"ON THIS FLOOR...":NC=NC+1
1645 HJ=HJ+1 IF HJ)=4 THEN HJ=0:FZ=1
1650 REM MORE CUSTOMERS STEPPING ON
1655 AA=RND(2) IF JV=1 THEN 1730
1660 IF FZ=1 OR FR=1 THEN AA=1:FZ=0
1670 IF AA=1 THEN 1690
1680 GOTO 1730
1690 YU=RND(5):YY=C:C=C+YU:C2=YU
1700 IF C)=59 THEN JV=1:C=C-C2:GOTO 1730
1710 REM GET DESTINATIONS
1720 GOTO 1800
1730 REM CONTINUE (IF NOT FINISHED)
1740 KG=0:IF GK)=C THEN 1940
1750 GOSUB 2190
1760 GOTO 800
1770 PRINT KG;"OF THE CUSTOMER(S) JUST"
1780 PRINT"STEPPED OFF OF MY ELEVATOR, ";
1785 PRINT N$;" ":GOSUB 2700
1790 GK=GK+KG:GOTO 1650
1800 REM YY MORE DESTINATIONS
1810 YU=ABS(C-YY):IF GK)C THEN 1940
1820 PRINT N$," YOU JUST HAD";YU;"MORE"
1830 PRINT"CUSTOMERS STEP ONTO THE"
1840 PRINT"ELEVATOR. THE DESTINATION(S):"
1850 KG=0:PRINT:FOR I=YY+1 TO C
1860 F=RND(PEEK(49)+1)
1865 IF F=TF THEN 1860
1870 IF F=1 THEN F$(1)="GROUND":FZ=1
```

```
1880 F(I)=F:NEXT:REM FLOORS
1890 REM TITLE OF FLOORS
1900 FOR I=YY+1 TO C
1910 G$(I)=F$(F(I)):NEXT
1920 FOR I=YY+1 TO C
1930 GOTO 690
1940 REM STOP / CONT
1950 PRINT:PRINT"YOU HAVE NOW COLLECTED: "
1960 PRINT USING W$;PA:PRINT:GOSUB 2190
1965 PRINT:GOSUB 2500
1970 IF JV=1 OR GK)=C THEN 1990
1980 GOSUB 2190:GOTO 800:REM ALL NOT OFF
1990 PRINT:GOTO 2300
2000 REM DATA FOR FLOORS
2010 DATA WOMEN'S,MEN'S,CHILDREN'S,INFANT'S
2020 DATA TOYS,CARPET,COMPUTERS
2030 DATA APPLIANCES,FURNITURE,HARDWARE
2040 DATA HOBBY,AUTO,OFFICE
2050 DATA BATHROOM,BEDROOM
2060 DATA CLOTHES,SUPPLIES,ITEMS
2070 REM DATA ELEMENT 17 WILL BE USED
2080 REM WITH ELEMENTS 2 - 5
2090 REM DATA ELEMENT 18 WILL BE USED
2100 REM WITH ELEMENTS 11 - 14
2110 REM DATA ELEMENT 19 WILL BE USED
2120 REM WITH ELEMENTS 15 & 16
2150 PRINT:PRINT"PRESS A KEY..."
2160 A$=INKEY$
2170 IF A$="" THEN 2160
2180 RETURN
2190 REM TIME LOOP
2200 FOR T=1 TO TT:NEXT
2210 GOTO 2180
2300 REM CONTINUE ?
2310 PRINT"WOULD YOU LIKE TO TAKE ON"
2320 PRINT"SOMEMORE CUSTOMERS ";N$
2330 PRINT"YOUR TOTAL EARNINGS:";
2340 PRINT USING W$;PA;
2350 INPUT A$
2360 IF A$()"Y" AND A$()"YES" THEN 2400
2370 TT=2000:JV=0:PRINT
2380 PRINT"YOU ARE ON FLOOR ";TF
2390 NC=0:GOTO 490
2400 PRINT
2410 PRINT"END OF ELEVATOR."
2420 END
2500 REM POINT SCALE / PAY DEDUCTION
2510 IF NC(=0 THEN NC=0:RETURN
```

```
2520 PRINT"YOU MADE";NC;"STOPS THAT WERE"
2530 PRINT"NOT NECESSARY ";N$;". YOU"
2540 PRINT"WILL TAKE A DEDUCTION IN PAY"
2550 PRINT"OF $20.00 FOR EACH ONE OF"
2560 PRINT"THOSE STOPS, TOTAL";
2570 NN=NC*20:PRINT USING W$;NN
2580 PRINT"LEAVING YOU";
2590 PA=PA-NN
2600 PRINT USING W$;PA:GOSUB 2190
2610 RETURN
2700 REM ANY TIPS ?
2710 IF KG(=1 THEN RETURN
2720 AA=RND(2)
2730 IF AA=1 THEN RETURN
2740 TI=RND(100)
2750 PA=PA+TI
2760 PRINT
2770 PRINT"YOU'RE IN LUCK ";N$;", A"
2780 PRINT"NICE CUSTOMER JUST GAVE"
2790 PRINT"YOU A TIP OF";
2800 PRINT USING W$;TI
2810 GOSUB 2190:RETURN
```

Variables and Strings

W$ - Used for all money amounts
N$ - First name of user
N - 0/1 variable for list of instructions
P$ - User's current occupation
F$(I) - Titles of floors
TT - Time (pause) loop
RX - For print statements from data list
C - Number of floors
GK - Number of correct stops
TF - Present floor number
F(I) - Number of each floor (random)
G$(I) - Title of floor (random)
FR - Input floor number
D$ - Direction, up or down
NN - For downward direction
V - 0/1 conditional variable for print statements
T$,H$ - For print statements
FF - Floor number
P,FS - Poke locations
PA - Amount of pay

KG - Correct floor
XX - 0/1 condition, correct floor
NC - Counter, unnecessary stops
HJ - Counter, user guessing
FZ - Used in conjunction with HJ
JV - Number of floors reached
AA - Random 1 or 2 for selection of more patrons getting on the elevator
YU - Random amount for more customers
YY - For extra print elements (more customers getting on elevator)
C2 - Equals the amount of YU. If C (floors) is greater than or equal to 59, C will equal C minus C2 (the amount selected)
YU - Used again as the ABS amount of C minus YY
TI - Random amount for any tips user might receive

Explanation of Program Lines

Line 20 establishes the format in which amounts of money will be printed.

Line 30 contains the dimension statements for the arrays containing the floors and their titles.

Lines 50-360 display the general instructions if they are needed.

Lines 370-480 prints all floor numbers and titles. These are only displayed for a predetermined amount of time (variable TT) so the user will be dependent upon his or her memory.

Line 490 initializes the variables C and GK so that if the program is played more than once, these will be reset to 0 (zero).

Lines 520-550 select a random number of customers, up to and including 10.

Lines 560-610 select C floor numbers at random. The IF F=1 THEN 590 in line 590 insures that floor 1 will not be chosen during the first selection process. The IF F=TF then 590 in line 595 insures that the floor the user is currently on will not be chosen at the beginning of any round of play.

Lines 620-650 select the titles of the floors. F$(F(I)), and assigns these to G$(I).

Lines 660-790 print all floor titles selected. The user will be allowed to study these titles for as long as he or she wishes.

Lines 810-850 ask the user to input a direction, either down <D> or up <U>. If the variable NN is equal to 1, the direction will be downward.

Lines 860-865 set the contents of D$ to A$, which is the direction the user entered.

Line 868 loops to a print statement in line 950 and then back to line 810, if the user does not select <D> or <U> for a direction.

Lines 870-940 ask the user for a floor number (line 890, variable FR).

Line 895 is used only at the beginning of play. If the user selects floor 1, program branches to a print statement in line 1280 and then back to line 800.

Line 900 insures that selected floor number is not less than 1 or greater than 16.

Line 905 branches to a print statement if the user inputs the current floor (TF) he or she is on.

Lines 910-990 test each direction and the value of each floor number to see if they are coordinated. If they are not valid, the program will branch to a suitable print statement.

Lines 1000-1060 are the print statements displayed if user tries to go below floor 1 or above floor 16.

Lines 1070-1470 are used for the movements of the elevator floor numbers to simulate the motion of the elevator. These lines will either move elevator down or up and poke the floor numbers in the video memory.

Lines 1500-1590 compare the current floor number with the floor number in memory F(I). If the selected floor does match, variable PA (for pay) is increased by 10; variable KG, for total correct floors, is increased by 1. Subscripted variable F(I) is then equal to 0 (zero), so that it is not matched again during the loop. Variable XX is then equal to 1 because a customer did want off on the inputted floor.

Lines 1600-1620 test the value of the variable XX. If it is equal to 0 (zero), no customers wanted off. If it is equal to 1, one or more customers did get off.

Lines 1630-1645 display the print statement indicating that no customers wanted off.

Lines 1650-1700 select a number of additional elevator customers if AA is equal to 1. If the variable JV is equal to 1, the number of customers (variable C) has reached a limit of 59, and no more customers will be permitted on the elevator. Line 1670 tests the amount of AA. If it is equal to 1, the program will branch to line 1690, where up to 5 more customers will be allowed onto the elevator.

Lines 1710-1720 branches to line 1800 for the additional (YY+1 TO C) destinations.

Lines 1730-1760 continues the program if there are still customers who want to go to a certain floor.

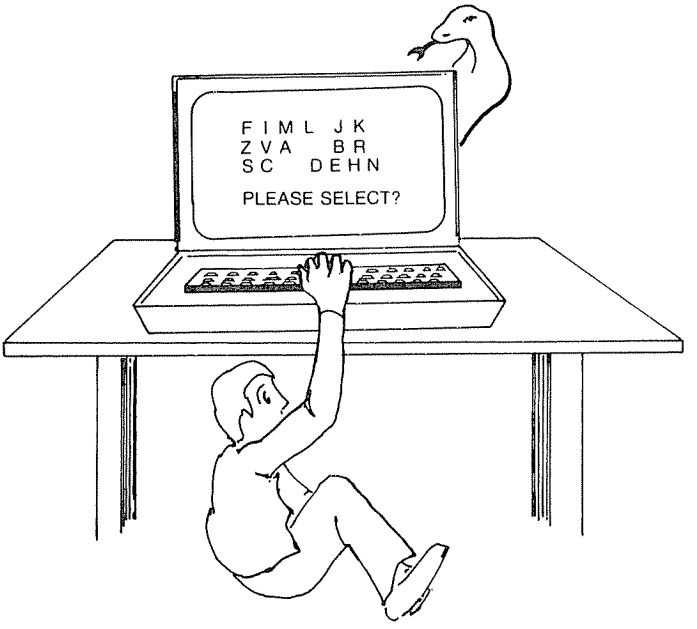
Lines 1770-1790 display the message telling the user how many customers got off of the elevator. Line 1785 branches to a GOSUB to see if the user might have received a tip.

Lines 1800-1840 display the message telling the user how many customers stepped onto the elevator.

Lines 1850-1930 select the number of floors and their titles. The program then branches back to line 690 for the printing of the titles. This loop begins at line 1920 and terminates at the next statement in line 650.

Lines 1940-1990 print the total amount the user has collected and then branch to line 2500 to see if the user had made any unnecessary stops. If he did, the total number of unnecessary stops is multiplied by 20 and deducted from his or her total pay. Line 1970 tests variables JV and GK. If neither conditions is met, the program resumes at line 800 (direction/floor entry). If either condition is met, the program branches to line 2300.

Lines 2300-2420 continue or stop the program, print the user's total earnings, and ask if he or she would care to earn more.

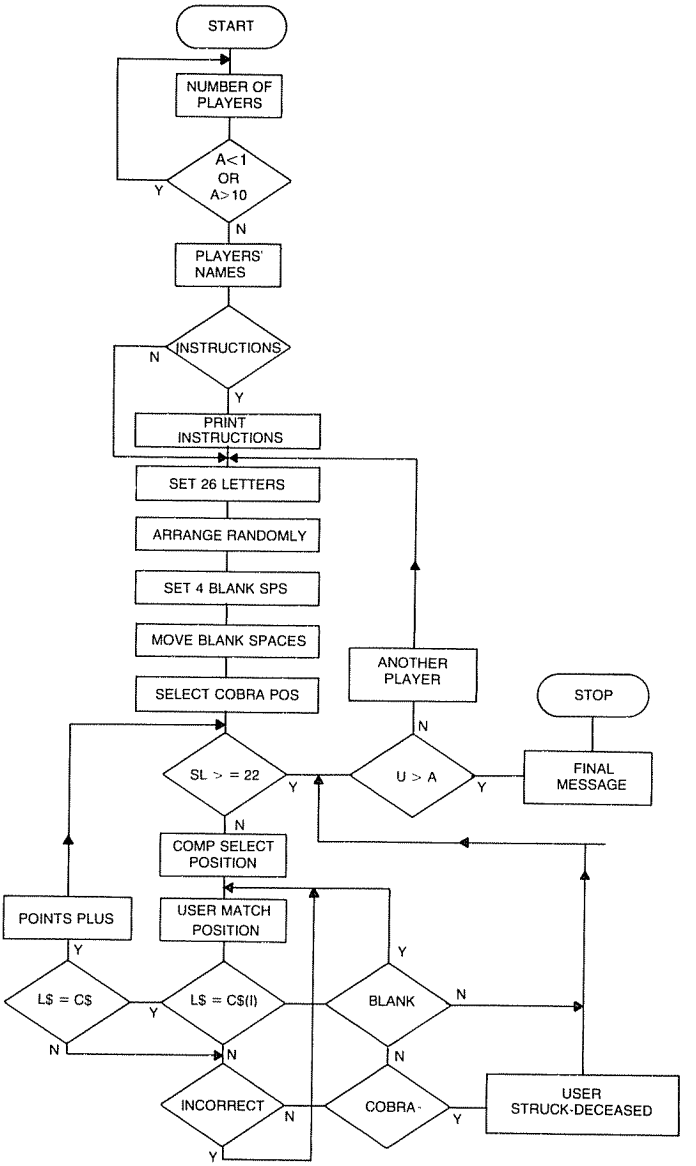


FAREWELL

With a quick eye you'll follow the asterisk to the computer's selected location within a 6 by 5 matrix of letters. All you'll need to do is tell the computer what letter location it landed on. Get it right and you'll gain 1000 points; get it wrong and you'll lose the same amount. Land on a cobra location...and you're dead! Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
***FAREWELL***

NUMBER PLAYING (1-10)? 2
THE 2 PLAYERS FIRST NAMES
? MAC
? SARAH

ANY OF YOU NEED TO SEE
THE INSTRUCTIONS? YES
```



Flowchart for *Farewell*.

R	H	P	V	U	J
N	Z	M	W	I	Q
	K	F		D	
G	B		A	S	O
C	X	L	Y	E	T

SELECT A POSITION
BY LETTER ONLY?—

PLAYER:

READ CAREFULLY, IT COULD MEAN
YOUR LIFE (OR THE GAME!!)...
EACH PLAYER WILL SEE AND STUDY A
MATRIX OF 6 BY 5 SQUARES OR
POSITIONS, THESE WILL BE LABELED WITH
THE LETTERS A THROUGH Z, WITH
4 SPACES BLANK. TO PLAY:
THE COMPUTER WILL SELECT 5
POSITIONS AND PLACE COBRAS
WITHIN EACH. THE COMPUTER WILL
SHOW YOU WHERE 3 OF THE COBRAS ARE
PRESS ENTER, PLEASE?

NOW THE COMPUTER WILL SELECT A
POSITION AND FLASH AN ASTERISK
THERE. WHAT EACH PLAYER MUST DO
IS ENTER THAT POSITION. IF YOU
ARE CORRECT, YOU'LL GAIN 1000
POINTS: IF INCORRECT, YOU'LL

LOSE 1000 POINTS. IF...YOU SELECT
A POSITION WHERE A COBRA IS...
YOU'RE OUT OF THE GAME, DEAD !!
SELECTION OF POSITIONS WILL CON-
TINUE FOR AS LONG AS EACH PLAYER
WANTS, OR 21 POSITIONS HAVE BEEN
OMITTED.....
PRESS ENTER, TO START?

Program Listing

```
 10 REM PROGRAM TITLE: FAREWELL
 15 CLS:RANDOM
 20 PRINT"*** FAREWELL ***"
 25 PRINT
 30 INPUT"NUMBER PLAYING (1-10)";A
 35 IF A<1 OR A>10 THEN 25
 40 PRINT"THE";A;"PLAYERS FIRST NAMES"
 45 FOR I=1 TO A
 50 INPUT Q$(I):NEXT
 55 PRINT:U=1
 60 PRINT"ANY OF YOU NEED TO SEE"
 65 INPUT"THE INSTRUCTIONS";X$
 70 IF X$<>"Y" AND X$<>"YES" THEN 210
 75 PRINT
 80 PRINT"READ CAREFULLY, IT COULD MEAN"
 85 PRINT"YOUR LIFE (OR THE GAME!!)..."
 90 PRINT"EACH PLAYER WILL SEE AND STUDY A"
 95 PRINT"MATRIX OF 6 BY 5 SQUARES OR"
100 PRINT"POSITIONS. THESE WILL BE LABELED WITH"
105 PRINT"THE LETTERS A THROUGH Z WITH"
110 PRINT"4 SPACES BLANK. TO PLAY:"
115 PRINT"THE COMPUTER WILL SELECT 5"
120 PRINT"POSITIONS AND PLACE COBRAS"
125 PRINT"WITHIN EACH. THE COMPUTER WILL"
130 PRINT"SHOW YOU WHERE 3 OF THE COBRAS ARE"
135 INPUT"PRESS ENTER, PLEASE";X$
140 PRINT:PRINT:PRINT@64,CHR$(30)
145 PRINT"NOW THE COMPUTER WILL SELECT A"
150 PRINT"POSITION AND FLASH AN ASTERISK"
155 PRINT"THERE, WHAT EACH PLAYER MUST DO"
160 PRINT"IS ENTER THAT POSITION. IF YOU"
165 PRINT"ARE CORRECT YOU'LL GAIN 1000"
170 PRINT"POINTS; IF INCORRECT YOU'LL"
175 PRINT"LOSE 1000 POINTS. IF...YOU SELECT"
180 PRINT"A POSITION WHERE A COBRA IS..."
```

```
185 PRINT"YOU'RE OUT OF THE GAME, DEAD !!"
190 PRINT"SELECTION OF POSITIONS WILL CON-"
195 PRINT"TINUE FOR AS LONG AS EACH PLAYER"
200 PRINT"WANTS, OR 21 POSITIONS HAVE BEEN"
205 PRINT"OMITTED....."
210 GOSUB 220:INPUT"PRESS ENTER, TO START";X$
215 CLS:GOTO 300
220 DIM L$(31),P(31),L(31)
225 REM LETTERS:? CHR$(23)
228 IF SL()0 PRINT"ARRANGING LETTERS..."
230 L=65:I=1:PZ=1000
235 L$(I)=CHR$(L):L(I)=I
240 IF I=27 THEN 255
245 I=I+1:L=L+1:GOTO 235
250 GOTO 235
255 REM SORT LETTERS RANDOMLY
260 J=1
265 N=RND(I-1)
270 IF L(N)=0 THEN 265
275 L$(J)=CHR$(N+64)
280 L(N)=0:J=J+1
285 IF J=I RETURN
290 GOTO 265
300 REM POSITIONS
305 P=10:I=1:PL=P+128:J=1
310 P(I)=P:L(I)=I
315 P=P+6:I=I+1:J=J+1
320 IF I=27 THEN 345
325 IF J=7 THEN 335
330 GOTO 310
335 P=PL:PL=PL+128:J=1
340 GOTO 310
345 REM THE REMAINING 4
350 L$(I)=" ":P(I)=P:L(I)=I
355 P=P+6:I=I+1:J=J+1
360 IF J=7 THEN 370
365 GOTO 350
370 REM MOVE THE 4 BLANKS
375 I=1:J=1:L=27
380 N=RND(26)
385 IF L(N)=0 THEN 380
390 F$=L$(N):L$(N)=L$(L)
395 L$(L)=F$:L(N)=0
400 L=L+1:IF L=31 THEN 410
405 GOTO 380
410 REM NOW PRINT LETTERS
415 CLS:FOR I=1 TO 30
420 PRINT@P(I),L$(I);:NEXT
```

```
425 REM FINAL GRAPHICS
430 X=16
435 FOR Y=0 TO 28
440 SET(X,Y):NEXT
445 X=X+12
450 IF X=100 THEN 460
455 GOTO 435
460 FOR X=16 TO 88
465 SET(X,Y):NEXT
470 Y=Y-6.5
480 IF Y=-3.5 THEN 490
485 GOTO 460
490 REM COBRA POSITIONS / PLAYER
495 C=1:PRINT@832,"PLAYER: ";Q$(U);
500 N=RND(I-5)
505 IF L(N)=0 OR L(N)=100 THEN 500
510 CB(C)=P(N):F$=L$(N):C$(C)=F$
515 L$(N)=CHR$(42):L(N)=100
520 IF C=5 THEN 560 ELSE IF C<=3 GOSUB 530
525 C=C+1:GOTO 500
530 XX=0:REM DEFINE 3 (1 FOR COMPUTER)
535 PRINT@P(N),L$(N);:FOR T=1 TO 100:NEXT T
540 PRINT@P(N),"  ";:FOR T=1 TO 50:NEXT T
545 PRINT@P(N),F$;
550 IF XX<=0 THEN XX=XX+1:GOTO 535
555 RETURN
560 REM COMPUTER SELECT (IF NOT DONE)
565 SL=1:W=X*7+25
575 IF SL>=22 THEN 1000 ELSE GOSUB 1200
585 PRINT@W,"I WILL SELECT A POSITION"
590 N=RND(L-1):GOSUB 1240
595 IF L(N)=0 OR L(N)=100 THEN 590
600 F$=L$(N):IF SL=1 THEN GOSUB 530
605 C$=F$:NU=N:GOSUB 1300:GOSUB 1200
610 PRINT@W,"SELECT A POSITION ";Q$(U);
615 PRINT@W+64,"BY LETTER ONLY";
620 INPUT L$
625 I=1
635 IF L$=L$(I) THEN 665
645 I=I+1
655 IF I>=31 THEN 675
660 GOTO 635
665 N=I:F$=L$:GOSUB 530
670 GOTO 800
675 REM COBRA POSITION OR BLANK
685 GOSUB 1200
690 FOR I=1 TO 5
695 IF L$=C$(I) THEN 735
```

```
705 NEXT
715 PRINT@W,"YOU CANNOT SELECT A BLANK !!"
725 GOSUB 1240:GOSUB 1200:GOTO 610
735 REM COBRA
745 PRINT@W,"YOUR DEAD ";Q$(U);" !!!"
755 PRINT@W+64,"THAT IS A COBRA POSITION !!"
765 PRINT@W+128,"** FAREWELL **":GOSUB 1240
770 ZZ=1:GOSUB 1200:GOTO 1020
800 GOSUB 1240:GOSUB 1200:IF L$()C$ THEN 900
805 PRINT@W,"CONGRATULATIONS ";Q$(U);"..."
810 PRINT@W+64,"YOU'VE JUST GAINED";PZ;
815 PRINT"POINTS !!"
820 REM POINTS / CONTINUE
825 W(U)=W(U)+PZ
830 PRINT@P(NU),"  ";
835 PRINT@872,"TOTAL POINTS: ";W(U);
840 GOSUB 1240:GOSUB 1200
845 PRINT@W,"YOU NOW HAVE";W(U);"POINTS..."
850 PRINT@W+64,Q$(U);"...CONTINUE (Y/N)";
855 INPUT X$:IF X$="N" THEN 1020
860 SL=SL+1:L(NU)=0:L$(NU)=" "
865 GOTO 575
900 REM WRONG POSITION
905 PRINT@W,"WRONG POSITION ";Q$(U)
910 PRINT@W+64,"YOU ARE NOW MINUS";PZ;
915 PRINT"POINTS...":GOSUB 1240
920 GOSUB 1200:W(U)=W(U)-PZ
925 PRINT@872,"TOTAL POINTS: ";W(U);
930 GOTO 610
1000 REM FINAL / ANOTHER PLAYER
1005 GOSUB 1240:GOSUB 1200
1010 PRINT@W,"THAT'S IT ";Q$(U);". YOU'VE"
1015 PRINT@W+64,"REACHED THE LIMIT..."
1020 PRINT@W+128,"ENTER...PLEASE";
1025 INPUT X$:CLS
1030 IF ZZ=1 THEN ZZ=0:GOTO 1040
1035 T(U)=W(U)
1040 U=U+1:IF U)A THEN 1100
1045 PRINT Q$(U);" YOU ARE NEXT..."
1050 PRINT"KEY ENTER WHEN READY";
1055 INPUT X$:CLS
1060 GOSUB 225:GOTO 300
1100 REM FINAL / SCORE RESULTS
1105 PRINT"PLAYERS AND POINTS:"
1110 FOR I=1 TO A
1115 PRINT Q$(I);" TOTAL POINTS: ";T(I)
1120 NEXT
1125 PRINT
```

```
1130 PRINT"PROGRAM TERMINATED....."
1135 END
1200 REM CLEAR PRINT AREA
1210 PRINT@W,CHR$(30)
1220 PRINT@W+64,CHR$(30)
1230 RETURN
1240 REM TIME DELAY
1250 FOR TI=1 TO 1200:NEXT
1260 RETURN
1300 REM FLASH SELECTION COURSE
1305 FOR I=1 TO NU
1310 IF L$(I)="*" OR L$(I)=" " THEN 1330
1315 PRINT@P(I),"*";
1320 FOR TI=1 TO 10:NEXT TI
1325 PRINT@P(I),L$(I);
1330 NEXT I
1335 RETURN
```

Variables and Strings

A - Total number playing (1-10)
U - Used for subscripted variables, strings, players and points
L$(I) - Letters, asterisks, blanks
P(I) - Position, Print @ areas
L(I) - For deletion of letters after use
SL - Number of letters used
L - For letters
PZ - For points
J - General purpose, random count of letters, positions
P - Positions, placed into subscripted variable P(I)
PL - To advance variable P two lines, when J= 7
N - Random selection variable
F$ - Contents of L$(I)
Q$(U) — Current player
C,CB(I),C$(I) - Cobra positions, a total of five
NU - Selection by computer for the location
ZZ - Indicates user killed by cobra, no points
W(U) - User's total score
T(U) - User's final score if ZZ = 0

Explanation of the Program Lines

Lines 30-55 allow up to 10 players' first names to be entered.

Lines 60-210 display the instructions, if they are needed by any player.

Lines 220-250 place the 26 letters of the alphabet in the subscripted string L$, and set the subscripted variable L to the value of I for random selection.

Lines 255-290 arrange the 26 letters randomly.

Lines 300-340 set the 26 positions at which the letters will be placed (Print @ areas).

Lines 345-365 set the remaining four spaces as blanks.

Lines 370-405 replace the four blank spaces with randomly selected letters. Where these letters were, there are now blank spaces.

Lines 410-420 print all letters and blank spaces.

Lines 425-485 produce the final X,Y graphics for the 6 by 5 matrix.

Lines 490-525 select five cobra positions and flash three of the positions with asterisks.

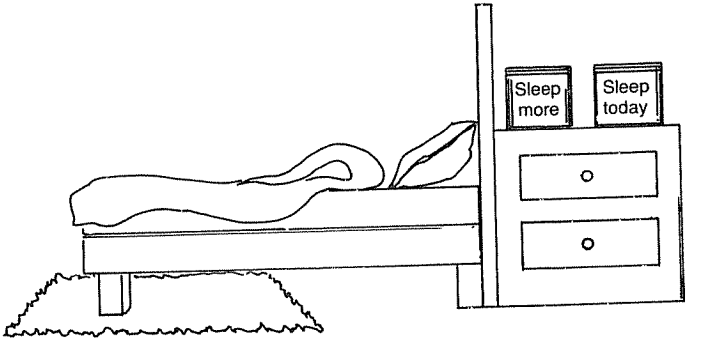
Lines 530-555 display the position selected by the computer and the selection given by the user.

Lines 560-605 cause the computer to select a random position, IF SL<=21. The first selection is shown by a flashing asterisk. The program then branches to lines 1300-1335 where asterisks are flashed rapidly from top position to the position selected by the computer.

Lines 610-1060 get the user's selection and determine if the entry is a correct guess, a blank, or a cobra position. If it is incorrect, the user will have 1000 points deducted from his or her score. If it is a blank, the user will be asked to try again. If it is a cobra, the user is automatically terminated. Correct entries that match the computers, are worth 1000 points each.

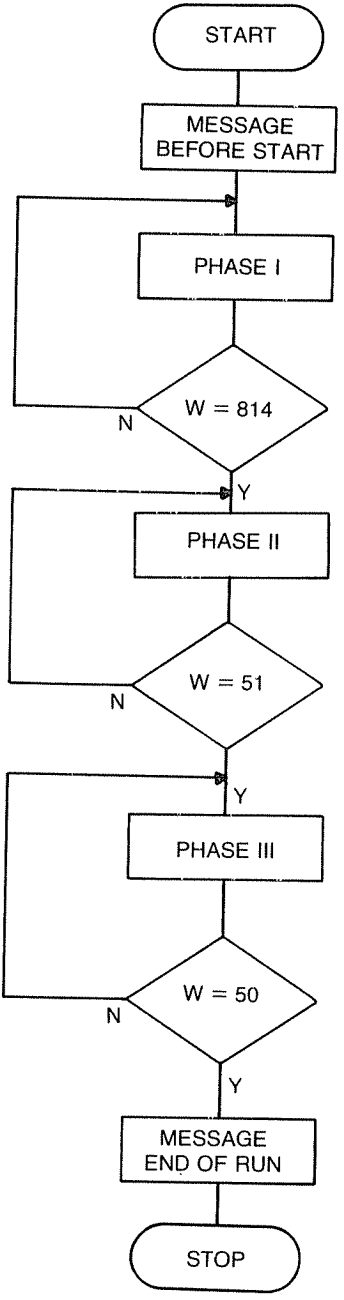
Lines 1100-1135 print all the players' names and their scores.

Lines 1200-1260 are routines used to clear Print @ areas for messages and to cause time delays for the reading of the messages.



INSOMNIA

Having trouble sleeping? The computer will try and solve that problem through using its graphics. This program is not intended to help people with severe sleep disorders nor should it be used by people who hypnotize easily. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.



Flowchart for *Insomnia*.

Program Listing

```
 10 REM PROGRAM TITLE: INSOMNIA
 20 CLEAR 200:CLS
 30 PRINT"INSOMNIA"
 40 PRINT
 50 PRINT"SO YOU ARE HAVING PROBLEMS"
 60 PRINT"SLEEPING. JUST SIT BACK, KEEP"
 70 PRINT"YOUR EYES ON THE VIDEO AND"
 80 INPUT"PRESS ENTER";X:CLS
 90 A=30:W=0
100 PRINT:PRINT TAB(A)":";
110 PRINT TAB(A-1)":";TAB(A+1)":";
120 I=RND(2)
130 IF I=1 THEN 160
140 IF A>=60 THEN 170
150 A=A+2:GOTO 180
160 IF A<=10 THEN 150
170 A=A-2
180 W=W+1:IF W=15 THEN 350
190 IF W=700 THEN 210
200 GOTO 100
210 IF W<=800 THEN 230
220 GOTO 250
230 PRINT
240 W=W+1:GOTO 210
250 PRINT"GETTING SLEEPY NOW ????"
260 FOR I=1 TO W:NEXT
270 IF W<=814 THEN 290
280 GOTO 300
290 PRINT:W=W+1:GOTO 270
300 FOR X=0 TO 127
310 SET(X,0):SET(X,47):NEXT
320 FOR X=1 TO 46
330 SET(0,X):SET(127,X):NEXT
340 FOR I=1 TO W:NEXT:W=0:GOTO 400
350 PRINT@990,;
360 PRINT"KEEP EYES HERE";
370 GOSUB 600
380 GOTO 190
400 L=62:A=65:X=897:Y=A
410 A$=STRING$(L,L*2+7)
420 PRINT@A,A$;
430 PRINT@X,A$;
440 A$=STRING$(L,32)
450 PRINT@A,A$;
460 PRINT@X,A$;
470 A=A+64:X=X-64:IF A=513 THEN 530
```

```
480 IF X=Y THEN 500
490 GOTO 410
500 IF W<=50 THEN 520
510 GOSUB 600:GOTO 610
520 W=W+1:IF W=25 GOSUB 580
525 GOTO 400
530 IF W>1 THEN 480
540 PRINT@A-45,;
550 PRINT"STARE AT THIS LOCATION";
560 GOSUB 600
570 GOTO 480
580 PRINT@A-429,;
590 PRINT"NOW FOLLOW THE BARS";
600 FOR I=1 TO 1000:NEXT:RETURN
610 CLS
620 PRINT"IF YOU ARE NOT GETTING SLEEPY"
630 PRINT"BY NOW, YOU'RE A HOPELESS"
640 PRINT"CASE !!!"
650 GOSUB 600:PRINT
660 PRINT"THIS IS THE LAST PHASE OF"
670 PRINT"INSOMNIA. IF IT FAILS, RUN"
680 PRINT"THE ENTIRE PROGRAM ONCE"
690 PRINT"AGAIN....."
700 GOSUB 600:PRINT
710 PRINT"KEEP YOUR EYES ON THE MOVING"
720 PRINT"BLOCK, THINKING TO YOURSELF"
730 PRINT"I AM GETTING SLEEPY....."
740 INPUT"KEY ENTER";X:W=0
750 CLS:P$=STRING$(2,191):A$=STRING$(2,32)
760 Y=61:FOR X=15 TO 26
770 SET(Y,X):NEXT
780 P=X+579
790 GOSUB 1100
800 FOR X=15 TO 26
810 RESET(Y,X):NEXT
820 IF Z=1 THEN 930
830 FOR X=15 TO 26
840 SET(Y,X):Y=Y-1:NEXT
850 GOSUB 1120
860 P=P-6
870 GOSUB 1100
880 Y=61
890 FOR X=15 TO 26
900 RESET(Y,X):Y=Y-1:NEXT
910 GOSUB 1120
920 Z=1:GOTO 760
930 GOSUB 1120
940 FOR X=15 TO 26
```

```
950 SET(Y,X):Y=Y+1:NEXT
960 P=P+5:PRINT@P,P$;
970 Y=61:FOR X=15 TO 26
980 RESET(Y,X)
990 Y=Y+1:NEXT
1000 GOSUB 1120
1010 W=W+1
1020 IF W=5 THEN 1050
1030 IF W=50 THEN 1150
1040 Z=0:GOTO 760
1050 FOR X=46 TO 75
1060 SET(X,14):SET(X,31):NEXT
1070 FOR X=14 TO 31
1080 SET(46,X):SET(75,X):NEXT
1090 GOTO 1030
1100 PRINT@P,P$;
1110 RETURN
1120 PRINT@P,A$;
1130 RETURN
1150 REM END OF PROGRAM
1160 GOTO 1160
```

Variables and Strings

A - Tab locations for first phase
W - General purpose counter
I - A random amount, also for a time pause
L - For putting character contents into a string
A,X - Print @ locations for second phase of program
Y - Contents of A for reset of graphics
A$ - Characters for graphics
P$,A$ - Characters for graphics for third phase of program
Y - For set/reset locations in third phase
P - Print @ locations (for block)
Z - 0/1 variable for graphics

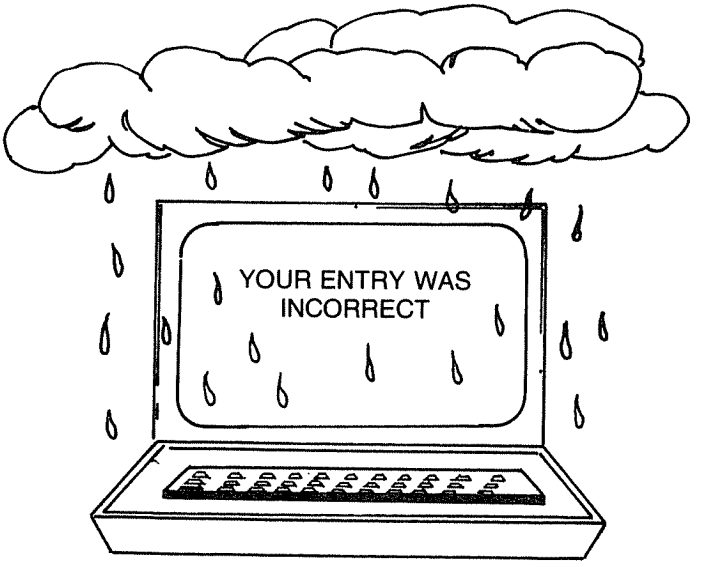
Explanation of the Program Lines

Lines 10-80 clear an amount of memory for graphic strings and display a short message before beginning the program.

Lines 90-290 form phase 1 of the program. They print three colons at a random tab location. These will move up in a snake like fashion.

Lines 300-590 form phase 2 of the program. Here, two bars move from the outside to the inside of a square, and then recycle.

Lines 610-1090 form the final phase of the program. A pendulum is set to swinging.

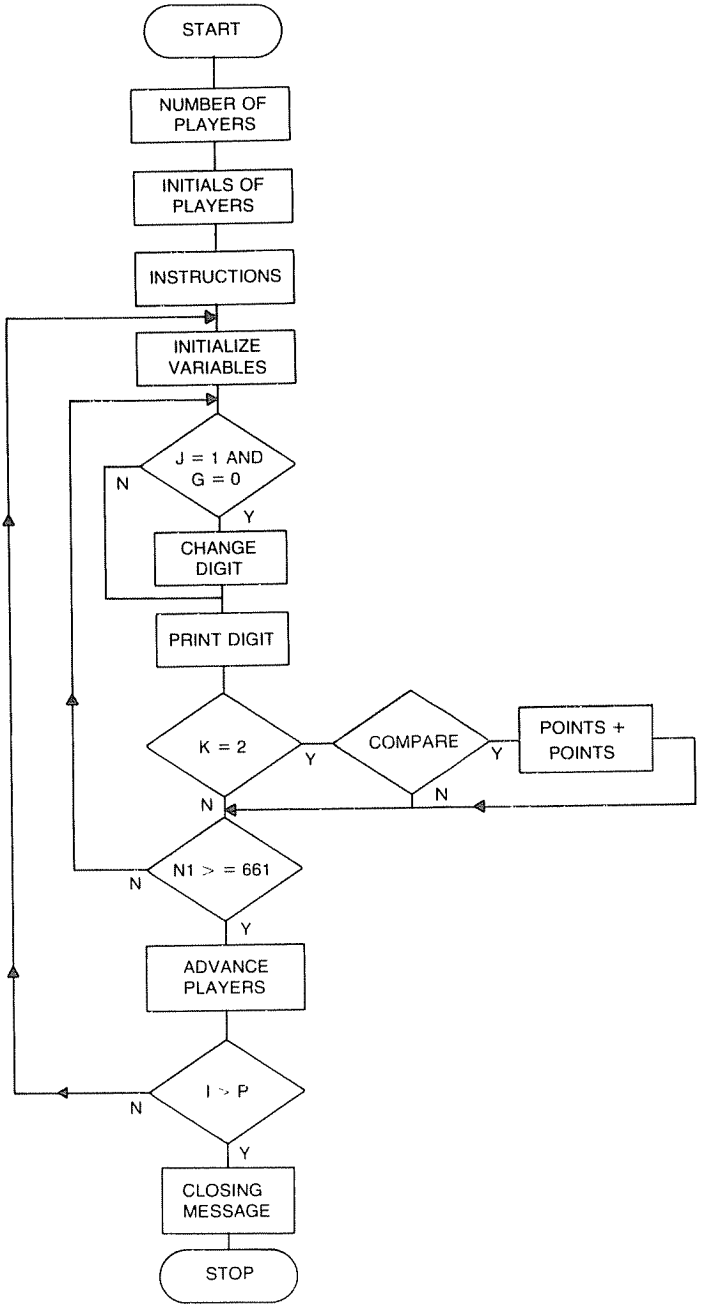


RAINY DAZE

This program is designed for bad weather days, to make them worse! All you'll need do is maintain good eye contact with the video and follow the numbers. When you see a number that is out of order, press the clear key. That's all there is to it. Oh yes, I almost forgot, the speed at which the numbers are printed across the screen gets faster and faster with each new line... Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

Sample Run

```
INSTRUCTIONS? YES
PLAYERS? 2
THE 2 PLAYER'S INITIALS
? D.F.
? H.J.

THIS PROGRAM IS DESIGNED TO
HELP YOU THROUGH YOUR WORST
DAYS BY MAKING THEM WORSE !!
ACTUALLY THIS PROGRAM WILL
```



Flowchart for *Rainy Daze*.

TEST YOUR ABILITY TO FOLLOW
NUMBERS IN SEQUENCE.
THE NUMERALS 1 THROUGH 660 WILL
RACE ACROSS AND FILL THE VIDEO.
WHEN A NUMERAL APPEARS THAT IS
NOT SUPPOSED TO BE THERE, HOLD
THE CLEAR KEY FOR A SHORT
PERIOD UNTIL THAT NUMERAL IS
BLOCKED OUT.

PRESS ENTER?

POINTS WILL BE GIVEN ON HOW
MANY OUT-OF-SEQUENCE NUMERALS
YOU HAVE DETECTED AT THE
END OF THE COUNT.
REMEMBER, YOU MUST HAVE GOOD
EYE POINT CONTACT WITH THE
VIDEO...

D.F. YOU WILL NOW PLAY.
PRESS ENTER, PLEASE?

Program Listing

```
 10 REM PROGRAM TITLE: RAINY DAZE
 20 CLS
 30 INPUT"INSTRUCTIONS";A$
 40 INPUT"PLAYERS";P:IF P=0 THEN 40
 50 PRINT"THE";P;"PLAYERS INITIALS"
 60 IF P>=11 THEN 20 ELSE FOR I=1 TO P
 70 INPUT N$(I):NEXT
 80 IF A$<>"Y" AND A$<>"YES" THEN 340
 90 REM INSTRUCTIONS
100 PRINT"THIS PROGRAM IS DESIGNED TO"
110 PRINT"HELP YOU THROUGH YOUR WORST"
120 PRINT"DAYS, BY MAKING THEM WORSE !!"
130 PRINT"ACTUALLY THIS PROGRAM WILL"
140 PRINT"TEST YOUR ABILITY TO FOLLOW"
150 PRINT"NUMBERS IN SEQUENCE."
160 PRINT"THE NUMERALS 1 THROUGH 660 WILL"
170 PRINT"RACE ACROSS AND FILL THE VIDEO."
180 PRINT"WHEN A NUMERAL APPEARS THAT IS"
190 PRINT"NOT SUPPOSED TO BE THERE, HOLD"
```

```
200 PRINT"THE CLEAR KEY FOR A SHORT"
210 PRINT"PERIOD UNTIL THAT NUMERAL IS"
220 PRINT"BLOCKED OUT."
240 PRINT
250 INPUT"PRESS ENTER";X
260 CLS
270 PRINT"POINTS WILL BE GIVEN ON HOW"
280 PRINT"MANY OUT-OF-SEQUENCE NUMERALS"
290 PRINT"YOU HAVE DETECTED AT THE"
300 PRINT"END OF THE COUNT."
310 PRINT"REMEMBER, YOU MUST HAVE GOOD"
320 PRINT"EYE POINT CONTACT WITH THE"
330 PRINT"VIDEO...":PRINT
340 I=1
350 IF P=1 THEN 390
360 PRINT N$(I);" YOU WILL NOW PLAY."
370 INPUT"PRESS KEY ENTER, PLEASE";X
380 GOTO 410
390 PRINT"WHEN READY, ";N$(P);" PRESS"
400 INPUT"THE ENTER KEY";X
410 CLS:RANDOM:N=0:N1=0:TT=250
420 A=0:M=55:L=64:G=0
430 K=PEEK(14440):GOTO 550
440 PRINT@A,N;
450 FOR T=1 TO TT+TT
460 NEXT:A=A+5
470 IF A>=M THEN 490
480 GOTO 430
490 A=L:M=M+64:L=L+64:TT=TT-25:G=0
500 IF L>=1023 THEN 520
505 IF TT<=100 THEN TT=100
510 GOTO 430
520 FOR T=1 TO 1000:NEXT
530 CLS
540 GOTO 420
550 REM CHANGE OR LEAVE
560 IF K=2 THEN NJ=1:GOTO 690
570 N=N+1
580 N1=N1+1:IF N1<=2 THEN 610
590 J=INT(RND(0)*2):IF G=1 THEN J=0:N=N1
600 IF J=1 THEN N=N+1:G=1
610 IF K=0 AND ABS(N-N1)<>0 THEN Y=Y+1
620 IF NJ=1 THEN NJ=0:GOTO 640
630 IF N1>=661 THEN 800 ELSE GOTO 440
640 N1=N1+1
650 PRINT@A-5,STRING$(5,143);
660 REM PAUSE BEFORE CONT.
670 FOR T=1 TO 100:NEXT:K=0
```

```
680 GOTO 440
690 IF ABS(N-N1)=0 THEN 710
700 Y=Y-1:GOTO 720
710 Y=Y+1
720 A1=PEEK(A-6+15360)
730 A2=PEEK(A-8+15360)
740 IF A1=143 AND A2=143 THEN 760
750 GOTO 620
760 CLS:Y=Y+2
770 PRINT"QUIT HOLDING DOWN THE CLEAR"
780 PRINT"KEY !! LET GO YOU DINGY !!
790 FOR T=1 TO 1200:NEXT:CLS:GOTO 620
800 REM PLAYER ADVANCE
810 YY(I)=Y:Y=0:CLS
820 I=I+1
830 IF I>P THEN N1=N1-1:GOTO 850
840 GOTO 360
850 REM FINAL
860 PRINT"PLAYERS AND THEIR POINTS"
870 PRINT"ARE AS FOLLOWS:"
880 FOR I=1 TO P
890 PRINT N$(I);" IDENTIFIED ";
900 IF YY(I)=0 PRINT"ALL NUMERALS.":GOTO 930
910 IF YY(I)<0 YY(I)=-YY(I)
920 PRINT N1-YY(I);"NUMERALS."
930 NEXT
940 PRINT
950 PRINT"END OF PROGRAM."
960 END
```

Variables and Strings

P - Number of players
N$(I) - Player's initials
I - Current player
N - Current digit
N1 - Counter to change number
TT - Time loop to cause a delay before the next number is printed
A - Print @ locations for digits
M,L - Largest Print @ areas (limits)
G - 0/1 for digit change
K - Peek location for key closure
J - Digit changed to
Y - Correct / incorrect entries
A1,A2 - For test to determine whether or not the user is constantly holding down the clear key
YY(I) - User's total points

Explanation of the Program Lines

Lines 10-330 accept the number of players and their initials and display the general instructions.

Lines 340-400 determine the current player.

Lines 410-420 initialize the variables.

Line 430 is used for user key closure.

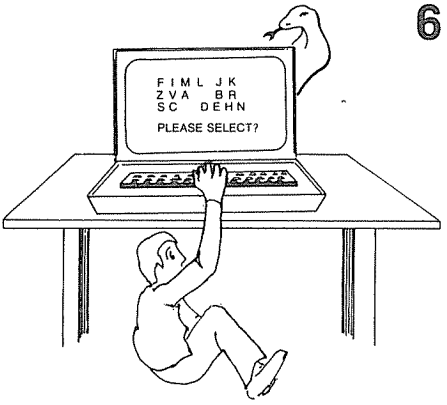
Lines 440-510 print the digits, changed or unchanged, test for a full video, and decrease the time loop.

Lines 520-540 create a pause before clearing the video.

Lines 550-680 leave current digit as it is or change it. If J is equal to 1 and G is equal to 0 (zero), the current digit will be changed.

Lines 690-790 increase the user's score if K is equal to 2 (clear key). They also test to see if the user is constantly holding down the clear key.

Lines 800-960 advance to next player or terminate the program and print the player's names and scores.



Games for Parties or Friendly Get-Togethers

Amount on Board is similar to roulette but uses squares rather than a wheel. Here's a way to experience the thrill of gambling at your own party.

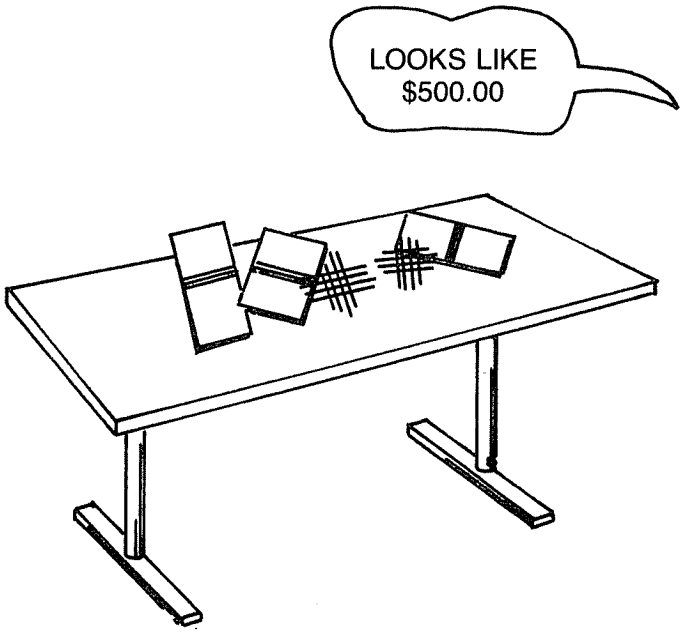
Lap-the-Track is another betting game. Whose bet will pay off in the great spot race?

Partners challenges you to guess a word based on clues your partner gives you.

From Point A to Point B is a different kind of maze game. You have to enter all your instructions before there is any movement at all.

Injection makes you responsible for killing the bacteria that are multiplying out of control.

Intercept commissions you to defend the domed city from alien intruders. Be careful. The high command disapproves when you kill friendly intruders.

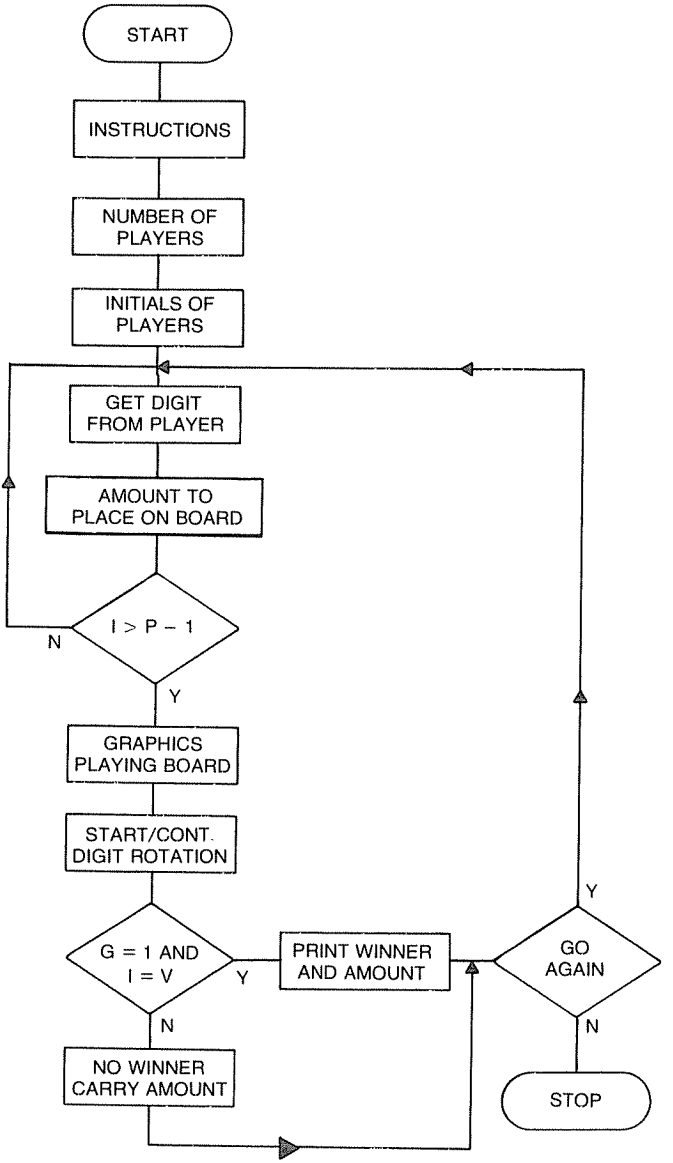


AMOUNT ON BOARD

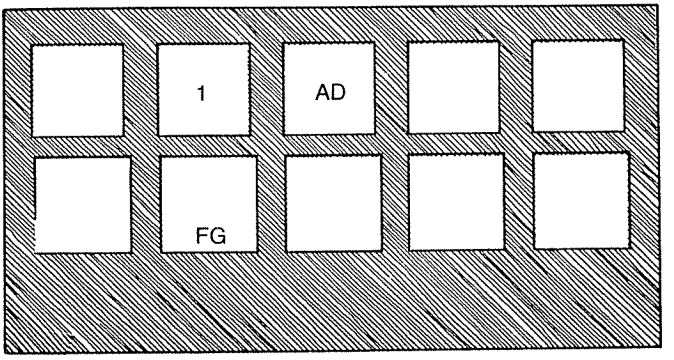
This game will allow you and up to 9 other players to place an amount of money on a square. These numbers will range from 0-9. Your initials will be placed on the numeral you have selected. If you plan to use real money, the computer will keep excellent tabs on all amounts. The computer also selects each number randomly, so it won't cheat anyone.

Sample Run

```
* AMOUNT ON BOARD *

INSTRUCTIONS BEFORE PLAY? YES

THIS IS THE FIRST PROGRAM
OF SIX THAT IS MADE FOR
PARTIES OR WHEN FRIENDS
GATHER LOOKING FOR SOMETHING
DIFFERENT TO DO. THIS GAME
IS SIMILAR TO ROULETTE, BUT
```



Flowchart for *Amount on Board*.



AMOUNT ON BOARD $20.00

WILL USE NO WHEEL. INSTEAD
10 SQUARES (FROM 0-9) WILL
BE USED. AT THE OPTION OF
THE PLAYERS, REAL MONEY
CAN BE USED. EACH PLAYER
(UP TO 10) WILL ENTER A

PRESS ENTER?

NUMBER FROM 0-9 AND PLACE ANY
BET (WITHIN REASON). WHAT-
EVER THE COMPUTER LANDS ON
DETERMINES THE WINNER. A
PLAYER CAN DROP OUT OR BE
ADDED DURING PLAY, BEFORE
ANY BETS ARE PLACED.

NUMBER OF PLAYERS? 2

INITIALS OF THE 2 PLAYERS
(WITHOUT SEPARATING PERIODS)
? AD
? FG
YOUR NUMBER SELECTION AD ?2
PLACE WHAT AMOUNT ON BOARD? 10

YOUR NUMBER SELECTION FG? 6
PLACE WHAT AMOUNT ON BOARD? 10
AMOUNT ON BOARD

Program Listing

```
100 REM PROGRAM TITLE: AMOUNT ON BOARD
110 CLEAR 200:RANDOM
120 CLS:M$="$$#,###.##":U=576
130 PRINT"* AMOUNT ON BOARD *"
140 PRINT
150 INPUT"INSTRUCT BEFORE PLAY";X$
160 IF X$="N" OR X$="NO" THEN 390
170 PRINT
180 PRINT"THIS IS THE FIRST PROGRAM"
190 PRINT"OF SIX THAT ARE MADE FOR"
200 PRINT"PARTIES OR WHEN FRIENDS"
210 PRINT"GATHER LOOKING FOR SOMETHING"
220 PRINT"DIFFERENT TO DO. THIS GAME"
230 PRINT"IS SIMILAR TO ROULETTE, BUT"
240 PRINT"WILL USE NO WHEEL. INSTEAD"
250 PRINT"10 SQUARES (FROM 0-9) WILL"
260 PRINT"BE USED. AT THE OPTION OF"
270 PRINT"THE PLAYERS, REAL MONEY"
280 PRINT"CAN BE USED. EACH PLAYER"
290 PRINT"(UP TO 10) WILL ENTER A"
300 PRINT
310 INPUT"PRESS ENTER";X$:CLS
320 PRINT"NUMBER FROM 0-9 AND PLACE ANY"
330 PRINT"BET (WITHIN REASON). WHAT-"
340 PRINT"EVER THE COMPUTER LANDS ON"
350 PRINT"DETERMINES THE WINNER. A"
360 PRINT"PLAYER CAN DROP OUT OR BE"
370 PRINT"ADDED, DURING PLAY, BEFORE"
380 PRINT"ANY BETS ARE PLACED.":PRINT
390 INPUT"NUMBER OF PLAYERS";P
400 IF P<1 OR P>10 THEN 390
410 PRINT"INITIALS OF THE";P;"PLAYERS"
415 PRINT"(WITHOUT SEPARATING PERIODS)"
420 FOR I=0 TO P-1:INPUT I$(I)
430 NEXT
440 REM GET NUMBER SELECT AND AMOUNT
450 FOR I=0 TO 9:N(I)=I:P(I)=I:X(I)=0:NEXT
460 N(0)=-99:I=0
470 PRINT"YOUR NUMBER SELECTION ";I$(I);
480 INPUT N:IF N<0 OR N>9 THEN 470
490 IF N(N)=0 THEN 510
495 IF N=0 THEN X(I)=N(0):GOTO 530
500 X(I)=N.GOTO 530
510 PRINT"NUMBER ALREADY ENTERED."
520 PRINT.GOTO 470
530 IF N<>0 THEN N(N)=0
```

```
540 PRINT"PLACE WHAT AMOUNT ON BOARD";
550 INPUT A(I)
560 IF A(I)<1 OR A(I)>100 THEN 580
570 GOTO 600
580 PRINT"KEEP THE BET WITHIN REASON."
590 GOTO 540
600 I=I+1
610 IF I>P-1 THEN 630
620 CLS:GOTO 470
630 REM TALLY AMOUNTS
640 I=0
650 A=A+A(I)
660 I=I+1:IF I>P-1 THEN 680
670 GOTO 650
680 REM INCREASE WITH LEFTOVER
690 A=A+M
700 M=A
710 A=0
720 CLS:PRINT CHR$(23)
730 REM PLAYING BOARD
740 B$=STRING$(21,143)
750 PRINT@4,B$;:PRINT@196,B$;
760 PRINT@388,B$,
770 J=4:JA=40 JB=J+64
780 PRINT@J,CHR$(191);
790 IF J>=JA THEN 810
800 J=J+8 GOTO 780
810 J=JB JA=JA+64:JB=JB+64
820 IF JA>=424 THEN 840
830 GOTO 780
840 REM LOCATIONS OF DIGITS
850 L=70
860 FOR I=0 TO 4:L(I)=L
870 L=L+8:NEXT
880 L=262
890 FOR I=5 TO 9:L(I)=L
900 L=L+8:NEXT:GOSUB 1600
910 TM=-10:PRINT@448,;
920 PRINT"AMOUNT ON BOARD";
930 PRINT USING M$;M
940 REM START THE ROUND
950 NI=RND(15)
960 I=0
970 PRINT@L(I),P(I);
980 IF G=1 AND I=V THEN 1140
990 IF I>=1 THEN 1040
1000 FOR Y=1 TO TM:NEXT
1010 I=I+1
```

```
1020 IF I=10 THEN 1040
1030 GOTO 970
1040 PRINT@L(I-1),"  ";
1050 IF I=10 THEN 960
1060 TM=TM+1.5
1070 IF TM>=80 THEN 1090
1080 GOTO 1000
1090 REM GET A NUMBER
1100 IF G=1 THEN 1080
1110 G=1
1120 V=INT(RND(0)*10)
1130 GOTO 1080
1140 REM WIN NUMBER
1145 IF I=0 PRINT@L(9),"  ";:GOTO 1155
1150 PRINT@L(V-1),"  ";
1155 FOR I=0 TO 9
1160 IF X(I)<>V THEN NEXT:GOTO 1230
1165 I$=I$(I):IF I$="" THEN 1230
1170 PRINT@U,;
1180 PRINT"THE WINNER ";I$
1190 PRINT"WINNING A TOTAL OF ";
1200 PRINT USING M$;M
1210 W(I)=W(I)+M:M=0
1220 GOTO 1270
1230 PRINT@U,;
1240 PRINT"SORRY, NO WINNER..."
1250 PRINT"THE MONEY STAYS IN THE"
1260 PRINT"POT UNTIL WON."
1270 REM RECYCLE
1280 PRINT:INPUT"PRESS ENTER, PLEASE";I$
1290 CLS:G=0
1300 PRINT"STOP OR CONTINUE PLAY (S/C)";
1310 INPUT I$
1320 IF I$="S" THEN 1530
1330 PRINT"ANY PLAYER WANT TO QUIT (Y/N)";
1340 INPUT I$
1350 IF I$="N" THEN 1445
1360 INPUT"YOUR INITIALS, PLEASE";I$
1370 FOR I=0 TO P-1
1380 IF I$(I)=I$ THEN 1400
1390 NEXT:GOTO 1445
1400 I$(I)="":REM SHUFFLE NAMES/AMOUNTS
1410 FOR T=0 TO P-1
1415 IF I$(T)="" THEN EX=1
1420 IF EX=0 THEN T$(T)=I$(T):T(T)=W(T):GOTO 1430
1425 T$(T)=I$(T+1):T(T)=W(T+1)
1430 NEXT:EX=0:FOR T=0 TO P-1
1435 I$(T)=T$(T):W(T)=T(T)
```

```
1440 NEXT:P=P-1
1445 IF P)=10 THEN 1510
1450 PRINT"ANY NEW PLAYERS (Y/N)";
1460 INPUT I$
1470 IF I$="N" THEN 1510
1480 REM INITIALS
1490 PRINT"YOUR INITIALS (WITHOUT PERIODS)";
1500 INPUT I$(P):P=P+1
1510 CLS
1520 GOTO 440
1530 REM ALL CHICKENED OUT
1540 CLS
1550 PRINT"PLAYERS AND AMOUNTS (YOU"
1560 PRINT"SHOULD HAVE ON YOU !!)"
1570 FOR I=0 TO P-1:PRINT I$(I);":";
1580 PRINT USING M$;W(I):NEXT
1590 PRINT:PRINT"END . .":END
1600 REM PRINT INITIALS AT LOCATION
1610 FOR I=0 TO 9
1620 IF X(I)=0 THEN 1650
1630 IF X(I)=-99 THEN 1660
1640 PRINT@L(X(I))+64,I$(I);
1650 NEXT:RETURN
1660 PRINT@L(0)+64,I$(I);
1670 X(I)=0:GOTO 1650
```

Variables and Strings

M$ - Print of dollar amount
P - Number of players
I$(I) - Player's initials
N(I) - User's number selection
P(I) - Digits 0-9
X(I) - User number selected
A(I) - User entered amount
A - Amount on board
M - Any leftover amount, if not won
B$ - For playing board (STRING$)
J,JA,JB - Locations for Print @ for playing board
L,L(I) - Location of the digits
TM - Pause between the printing of the digits
G,V - Computer select of digit
W(I) - User's winnings

Explanation of the Program Lines

Lines 100-380 display the general instructions.

Lines 390-400 accept the number of players.

Lines 410-430 accept the initials of the players, without separating periods. This is required so that the users' initials can be printed in the numeral blocks they have selected. If the user adds the separating periods, part of the graphics will be lost.

Lines 440-460 set subscripted variables to the numerals 1 through 9.

Lines 470-530 allow each user to input a number between 0 and 9.

Lines 540-620 allow each user to input an amount of money to place on board (any amount within reason).

Lines 630-710 compile all amounts to determine the total on board. The variable M will be an amount that is carried over from a play nobody won.

Lines 720-830 display the playing board.

Lines 840-900 determine the locations of the digits 0-9. Each location selected by a user will show his or her initials.

Lines 910-1080 form the main part of the program. All digits are flashed from left to right until TM>= 80. Then the computer stops at its preselected location.

Lines 1090-1130 select a random number.

Lines 1140-1220 print the winner, if there is one, and the amount he or she has won.

Lines 1230-1280 are used when there is no winner. The amount on board will remain in the pot until it is won (variable M).

Lines 1300-1320 allow any player to terminate the program.

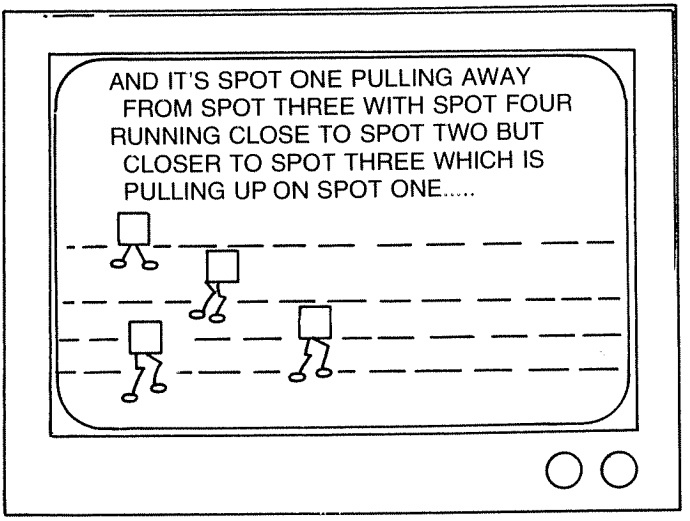
Lines 1330-1440 allow any player to drop out of the game if he or she so wishes. If a player does drop out, the names will be re-organized so they do not include the name of the player that dropped out.

Lines 1445-1500 allow a new player to join the game, if there are less than 10 players.

Lines 1510-1520 recycle the program with the new number selections and bets.

Lines 1530-1590 display a message when all players have quit message and also show the amount each has won.

Lines 1600-1670 form a subroutine that prints the initials of the players at the numeral they have selected.

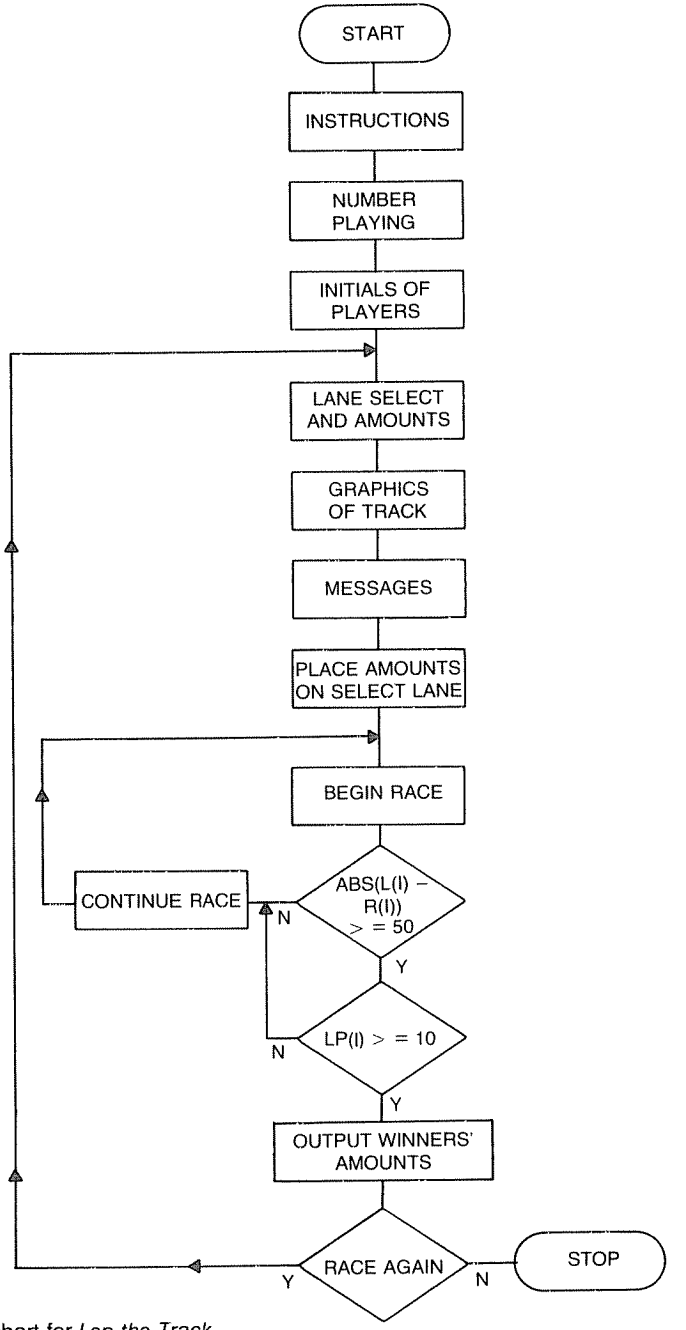

AND IT'S SPOT ONE PULLING AWAY
FROM SPOT THREE WITH SPOT FOUR
RUNNING CLOSE TO SPOT TWO BUT
CLOSER TO SPOT THREE WHICH IS
PULLING UP ON SPOT ONE.....

LAP-THE-TRACK

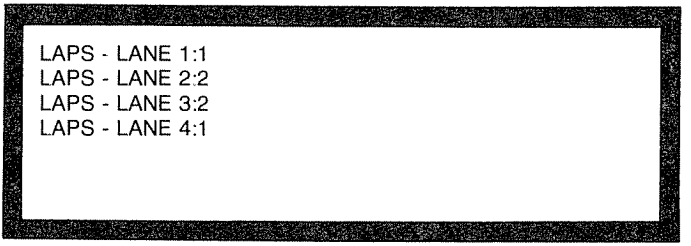
You and 14 others can bet on one of four spots that will dash across your video in a race that will last 10 laps. There are programs of horse races, dog races, etc., so, you could call this a spot race. After selecting a lane and placing your bet, the four spots will begin darting across the video. The winner, naturally, will be the spot that completes the 10 laps first. Each spot will have an equal opportunity to win, dependent upon the random function. All monies are divided equally should more than one person bet on the winning lane. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

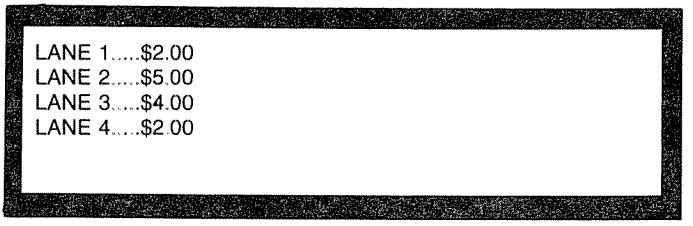
Sample Run

```
-:-: LAP-THE-TRACK :-:-

INSTRUCTIONS? YES

UP TO FIFTEEN PEOPLE BET ON
THE FOUR MOVING SPOTS THAT WILL
BE IN A FAST RACE TO THE FINISH.
A COMPLETE RACE CONSISTS OF
10 LAPS. EACH PLAYER WILL BE
ABLE TO CHOOSE A LANE (1-4)
```



Flowchart for *Lap-the-Track*.





AND HOW MUCH THEY WANT TO BET
ON THE CHOSEN LANE. IF THERE
ARE NO WINNERS, THE COMPUTER
WILL PLACE THE MONEY ON A
LANE OF ITS CHOICE IN THE

PRESS ENTER?

NEXT RACE. IF THE COMPUTER'S SPOT
(LANE) HAPPENS TO WIN, IT WILL
NOT TAKE THE MONEY, IT WILL
JUST BET ON ANOTHER LANE.
SO NO MATTER HOW YOU SLICE IT,
SOMEONE IS BOUND TO WIN (SO
YOU CAN USE REAL MONEY IN
THIS GAME TOO).
HOW MANY TO PLAY? 4

INITIALS, PLEASE? A.A.
? B.B.
? C.C.
? D.D.

YOU WILL NOW SELECT A LANE.
A.A. LANE NUMBER? 1
BET WHAT AMOUNT? 2

B.B. LANE NUMBER? 4
BET WHAT AMOUNT? 2

C.C. LANE NUMBER? 3
BET WHAT AMOUNT? 4

D.D. LANE NUMBER? 2
BET WHAT AMOUNT? 5

IF MORE THAN ONE PLAYER WINS
(SELECTION OF THE SAME LANE)
ALL MONEY WILL BE DIVIDED AS
EQUALLY AS POSSIBLE.

PRESS ENTER TO START THE RACE?

Program Listing

```
 10 REM PROGRAM TITLE: LAP-THE-TRACK
 20 CLEAR 200:RANDOM:CLS
 30 PRINT"-:-: LAP-THE-TRACK .-:-"
 40 PRINT:DIM A$(15),A(15),LA(15),T(15)
 50 INPUT"INSTRUCTIONS";A$
 60 IF A$<>"Y" AND A$<>"YES" THEN 300
 70 PRINT"UP TO FIFTEEN PEOPLE BET ON"
 80 PRINT"THE FOUR MOVING SPOTS THAT WILL"
 90 PRINT"BE IN A FAST RACE TO THE FINISH."
100 PRINT"A COMPLETE RACE CONSISTS OF"
110 PRINT"10 LAPS. EACH PLAYER WILL BE"
120 PRINT"ABLE TO CHOOSE A LANE (1-4)"
130 PRINT"AND HOW MUCH THEY WANT TO BET"
140 PRINT"ON THE CHOSEN LANE. IF THERE"
150 PRINT"ARE NO WINNERS, THE COMPUTER"
160 PRINT"WILL PLACE THE MONEY ON A"
170 PRINT"LANE OF ITS CHOICE IN THE"
180 PRINT:INPUT"KEY ENTER";A$
190 CLS
200 PRINT"NEXT RACE. IF THE COMPUTERS SPOT"
210 PRINT"(LANE) HAPPENS TO WIN, IT WILL"
220 PRINT"NOT TAKE THE MONEY, IT WILL"
230 PRINT"JUST BET ON ANOTHER LANE."
```

```
240 PRINT"SO NO MATTER HOW YOU SLICE IT,"
250 PRINT"SOMEONE IS BOUND TO WIN (SO"
260 PRINT"YOU CAN USE REAL MONEY IN"
270 PRINT"THIS GAME TOO )"
300 PRINT
310 INPUT"HOW MANY TO PLAY";A
320 IF A<1 OR A>15 THEN 320
330 PRINT"INITIALS, PLEASE";
340 FOR I=1 TO A
350 INPUT A$(I):NEXT
360 PRINT
370 PRINT"YOU WILL NOW SELECT A LANE."
380 I=1
390 PRINT A$(I);" LANE NUMBER";
400 INPUT LA(I)
410 IF LA(I)<1 OR LA(I)>4 THEN 390
420 PRINT"BET WHAT AMOUNT";
430 INPUT A(I)
440 I=I+1
450 IF I>A THEN 480
460 PRINT
470 GOTO 390
480 CLS:GOSUB 1300:IF ZZ=1 THEN 530
490 PRINT"IF MORE THAN ONE PLAYER WINS"
500 PRINT"(SELECTION OF THE SAME LANE)"
510 PRINT"ALL MONEY WILL BE DIVIDED AS"
520 PRINT"EQUALLY AS POSSIBLE."
530 PRINT:IF CB=1 THEN 960
540 PRINT"PRESS ENTER TO START THE RACE";
550 INPUT A$:CLS
560 REM OUTER OF TRACK
570 FOR X=0 TO 127
580 SET(X,0):SET(X,47)
585 SET(X,15):SET(X,31):NEXT
590 FOR X=1 TO 15
600 SET(0,X):SET(127,X):NEXT
610 FOR X=31 TO 47
620 SET(0,X):SET(127,X):NEXT
630 REM SPOT POSITIONS / MESSAGES
640 GOSUB 1200:GOSUB 1410:L=1:I=384
650 L(L)=I:R(L)=I:P(L)=I-300:LP(L)=0
660 I=I+64:IF L>=4 THEN 680
670 L=L+1:GOTO 650
680 FOR I=1 TO 4
690 PRINT@L(I),CHR$(143);:NEXT
700 REM START RACE
710 LP=0:X$=CHR$(143):PK=L(1)+63
720 I=RND(4):M=ABS(I-5)
```

```
730 PRINT@L(I)," ";
740 IF ABS(L(I)-R(I)))=50 THEN 1000
750 L(I)=L(I)+RND(14)
760 PRINT@L(I),X$;
770 IF M+1<>I AND M+1<>5 THEN 790
780 GOTO 820
790 PRINT@L(M+1)," ";
800 L(M+1)=L(M+1)+RND(4)
810 PRINT@L(M+1),X$;
820 IF M-1<>I AND M-1<>0 THEN 840
830 GOTO 870
840 PRINT@L(M-1)," ";
850 L(M-1)=L(M-1)+RND(4)
860 PRINT@L(M-1),X$,
870 IF M+2<>5 AND M+2<>6 THEN 890
880 GOTO 720
890 PRINT@L(M+2)," ";
900 L(M+2)=L(M+2)+RND(4)
910 PRINT@L(M+2),X$,
920 PRINT@L(M)," ";
930 L(M)=L(M)+RND(4)
940 PRINT@L(M),X$;
950 GOTO 720
960 CB=0:PRINT"THE COMPUTER WILL PLACE"
970 PRINT"THE AMOUNT OF";USING AA$;AM
980 RF=RND(4).PRINT"ON LANE";RF;"
985 MN(RF)=MN(RF)+AM:AM=0
990 GOTO 540
1000 PRINT@L(I)," ";
1010 L(I)=R(I)
1020 LP(I)=LP(I)+1:PRINT@P(I),LP(I);
1030 IF LP(I))=10 THEN 1500
1040 GOTO 720
1200 PRINT@66,"LAPS - LANE 1:";
1210 PRINT@130,"LAPS - LANE 2:";
1220 PRINT@194,"LAPS - LANE 3:";
1230 PRINT@258,"LAPS - LANE 4:";
1240 REM AMOUNTS PER LANE
1250 PRINT@706,"LANE 1.....";
1260 PRINT@770,"LANE 2.....";
1270 PRINT@834,"LANE 3.....";
1280 PRINT@898,"LANE 4.....";
1290 RETURN
1300 REM AMOUNTS ON LANES
1310 T=1
1320 FOR I=1 TO A
1330 IF LA(I)=T THEN 1350
1340 NEXT:GOTO 1370
```

```
1350 MN(T)=MN(T)+A(I)
1360 GOTO 1340
1370 T=T+1
1380 IF T>=5 THEN 1400
1390 GOTO 1320
1400 RETURN
1410 REM PRINT AMOUNTS
1420 AA$="$$#,###.##"
1430 I=1:FF=718
1440 PRINT@FF,USING AA$;MN(I);
1450 I=I+1:FF=FF+64
1460 IF I>=5 RETURN
1470 GOTO 1440
1500 REM FINISH
1510 WW=1
1520 PRINT@P(I)+6,"WINNER";
1530 FOR TU=1 TO 200:NEXT
1540 PRINT@P(I)+6,STRING$(6,32);
1550 FOR TU=1 TO 50:NEXT
1560 IF WW>=6 THEN 1580
1570 WW=WW+1:GOTO 1520
1580 REM PLAYER LANE SELECT
1590 WW=0
1600 FOR T=1 TO A
1610 IF LA(T)=I THEN 1630
1620 NEXT:GOTO 1650
1630 WW=WW+1:T(WW)=T
1640 GOTO 1620
1650 FOR I=1 TO 4:AM=AM+MN(I):MN(I)=0
1660 NEXT:IF WW=0 THEN 1760
1670 PRINT@735,"WINNER(S):";
1680 WX=799
1690 FOR I=1 TO WW
1700 PRINT@WX,A$(T(I));
1710 IF WW>1 PRINT",";
1720 WX=WX+5:NEXT
1725 IF WW=1 THEN 1790
1730 PRINT@863,"EACH WINNER RECEIVES:";
1740 PRINT@927,USING AA$;AM/WW;
1750 AM=0:GOTO 1820
1760 PRINT@P(1)+15,;
1770 PRINT"** NO WINNERS **";
1780 GOTO 1870
1790 PRINT@863,;
1800 PRINT"YOU WILL RECEIVE:";
1810 GOTO 1740
1820 REM ANOTHER
1830 PRINT@384,;
```

```
1840 INPUT"ANOTHER RACE (Y/N)";A$
1850 IF A$="N" THEN 2000
1860 CLS:ZZ=1:GOTO 360
1870 FOR TU=1 TO 1200:NEXT
1880 PRINT@P(1)+12,;
1890 PRINT"IF ANOTHER RACE IS HELD, THE";
1900 PRINT@P(2)+12,;
1910 PRINT"COMPUTER WILL BET THE MONEY";
1920 PRINT@P(3)+12,;
1930 PRINT"ON A LANE IT SELECTS.";
1940 CB=1:GOTO 1820
2000 REM END
2010 CLS
2020 PRINT"FAREWELL 'TIL THE NEXT RACE."
2030 END
```

Variables and Strings

A - Number playing (1-15)
A$(I) - Initials of players
LA(I) - Lane number selected by the user
CB - 0/1 to stop or race again
L(L) - Spot positions, start/continue
R(L) - Spot positions, greatest amount
P(L),LP(L) - Win location
M - Distance to move
RF - Lane selected by the computer if required
MN(I) - Total amount in bets
AM - Amount carried over from previous race
AA$ - Amount messages
FF - Print @ location for message

Explanation of Program Lines

Lines 10-270 display the general instructions.

Lines 300-360 accept the number of players and their initials.

Lines 370-470 allow each player to select a lane (1-4) and enter the amount he or she wants to bet.

Line 480 is explained at lines 1300-1400.

Lines 490-550 display a message concerning the betting.

Lines 560-620 draw the outer boundaries of the track.

Lines 640-690 establish the locations of the spot and display messages.

Lines 700-950 print all four spots and keep them moving to right. Random amounts are added to each, dependent upon variable

M. Laps are increased by one for each spot when ABS(L(I)–R(I))>= 50. The first to reach 10 laps is the actual winner.

Lines 960-990 allow the computer to select a lane and an amount to bet.

Lines 1000-1040 work in conjunction with lines 700-950, clearing the screen and counting the laps.

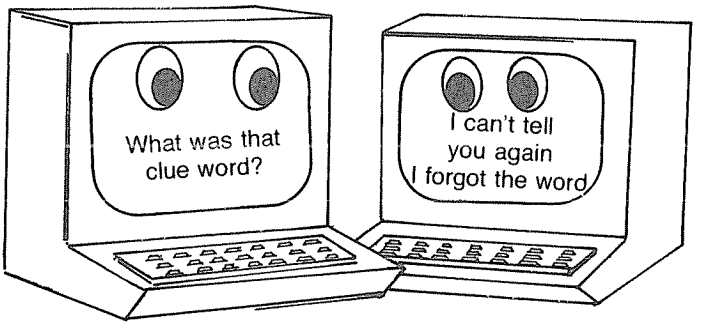
Lines 1200-1290 display messages that appear during the race.

Lines 1300-1400 tally the amounts bet on each lane MN(T) using the individual bets A(I).

Lines 1410-1470 print all bets as the race begins. The amount of each bet is placed beside the track that it was bet on.

Lines 1500-1810 print the names of all the winners and how much they won. If there is no winner, the amount bet will be carried over. If more than one player wins on the same track, the total amount bet will be divided equally between those players.

Lines 1820-2030 ask if the users want to race again and recycle the program or terminate appropriately.

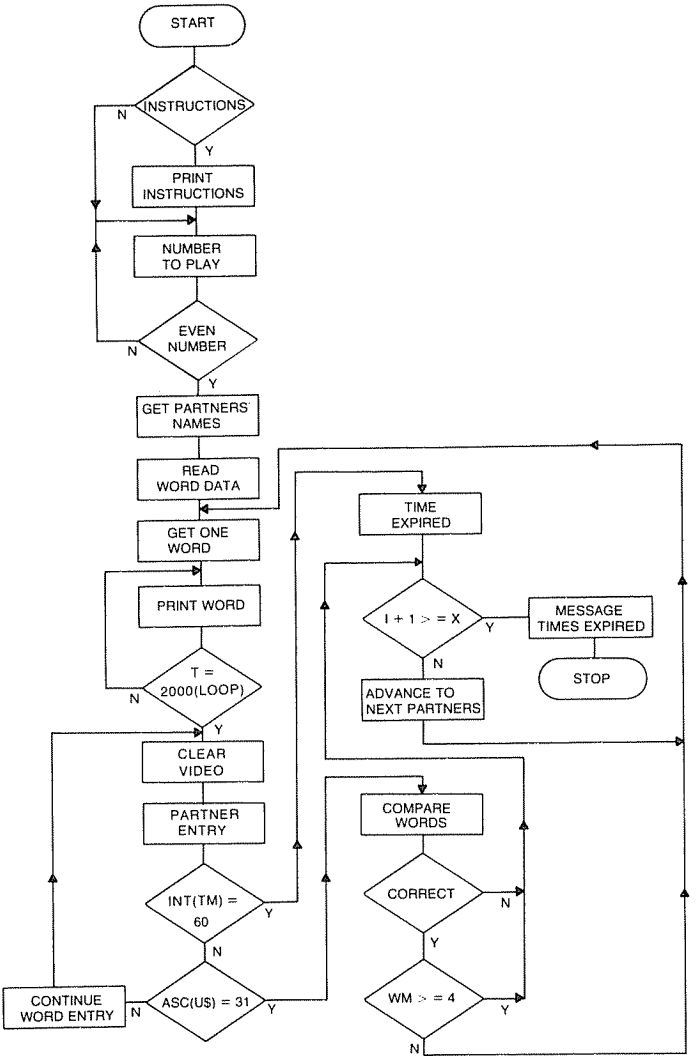


PARTNERS

How well you depend upon your partner will be determined during this game. Up to 10 players (5 partners) can compete against one another. There will be 5 words. One player will view the word; the video will clear; then the other player takes the keyboard. The first player, who viewed the word, will give his or her partner clues as to what the word is. One minute will be given to enter all 5 point words. The winners are the partners who get the most words correct.

Sample Run

```
PARTNERS

INSTRUCTIONS BEFORE PLAY? YES

HOW MUCH DO YOU RELY ON YOUR
PARTNER? IF YOU RELY ON HIM
OR HER TO THE FULLEST, THIS
GAME WILL SURE TELL. THIS
GAME REQUIRES AT LEAST 2
PLAYERS; WHILE UP TO 10 CAN
PLAY.
THE OBJECT OF THE GAME IS TO
INPUT 5 WORDS BEFORE A MINUTE
HAS ELAPSED. EACH TEAM WILL
SELECT WHO WILL SEE THE
WORD. AFTER SEEING IT THE PLAYER
WILL PRESS ENTER. THE SCREEN
WILL CLEAR.
```



Flowchart for *Partners*.

PRESS A KEY

YOUR PARTNER (WHO SAW THE
WORD) WILL THEN GIVE YOU CLUE
WORDS TO WHAT IT MIGHT BE, BUT
NOT THE ACTUAL WORD ITSELF.
YOU WILL THEN START INPUTTING
GUESSES OF WHAT YOU THINK THE
WORD IS. AS SOON AS YOU HAVE
INPUT THE CORRECT WORD,
TIMING WILL STOP. YOU AND YOUR
PARTNER WILL CHANGE PLACES,
CHANGE PLACES,
AND YOUR PARTNER
WILL VIEW
ANOTHER WORD, UP
TO A TOTAL OF 5.

PRESS A KEY.....

REMEMBER THAT YOU'LL ONLY HAVE
ONE MINUTE TO ENTER ALL 5 WORDS
(EXCEPT FOR CHANGING TO REVIEW
ANOTHER WORD).

NUMBER PLAYING? 5
EVEN NUMBER ONLY...

NUMBER PLAYING? 4
NOW ENTER THE 4 PEOPLE
THAT WILL BE PARTNERS,
AS EACH WILL PLAY.
? EARTHA
? FRAN

? WILL
? RANDY

Program Listing

```
10 REM PROGRAM TITLE: PARTNERS
20 CLEAR 500:CLS
30 RANDOM:DIM W$(50),W(50),T(11)
```

```
 40 PRINT"PARTNERS"
 50 PRINT
 60 INPUT"INSTRUCTIONS BEFORE PLAY (Y/N)";X$
 70 IF X$="N" OR X$="NO" THEN 470
 80 PRINT"HOW MUCH DO YOU RELY ON YOUR"
 90 PRINT"PARTNER? IF YOU RELY ON HIM"
100 PRINT"OR HER TO THE FULLEST, THIS"
110 PRINT"GAME WILL SURE TELL  THIS"
120 PRINT"GAME REQUIRES AT LEAST 2"
130 PRINT"PLAYERS, WHILE UP TO 10 CAN"
140 PRINT"PLAY."
150 PRINT"THE OBJECT OF THE GAME IS TO"
160 PRINT"INPUT 5 WORDS BEFORE A MINUTE"
170 PRINT"HAS ELASPED  EACH TEAM WILL"
180 PRINT"SELECT WHO WILL SEE THE "
190 PRINT"WORD  AFTER SEEING IT THE PLAYER"
200 PRINT"WILL PRESS ENTER; THE SCREEN"
210 PRINT"WILL CLEAR."
220 FOR T=1 TO 5000:NEXT :PRINT
230 PRINT"PRESS A KEY"
240 X$=INKEY$:IF X$="" THEN 240
250 PRINT@0,CHR$(30)
260 PRINT"YOUR PARTNER (WHO SAW THE "
270 PRINT"WORD) WILL THEN GIVE YOU CLUE"
280 PRINT"WORDS TO WHAT IT MIGHT BE, BUT"
290 PRINT"NOT THE ACTUAL WORD ITSELF."
300 PRINT"YOU WILL THEN START INPUTTING"
310 PRINT"GUESSES OF WHAT YOU THINK THE "
320 PRINT"WORD IS  AS SOON AS YOU HAVE"
330 PRINT"ENTERED THE CORRECT WORD"
340 PRINT"TIMING WILL STOP  YOU AND YOUR"
350 PRINT"PARTNER WILL CHANGE PLACES,"
360 PRINT"AND YOUR PARTNER WILL VIEW"
370 PRINT"ANOTHER  WORD, UP"
380 PRINT"TO A TOTAL OF 5."
390 PRINT
400 PRINT"PRESS A KEY....."
410 X$=INKEY$:IF X$="" THEN 410
420 CLS
430 PRINT"REMEMBER THAT YOU'LL ONLY HAVE"
440 PRINT"ONE MINUTE TO ENTER ALL 5 WORDS"
450 PRINT"(EXCEPT FOR CHANGING TO"
460 PRINT"REVIEW ANOTHER WORD.)"
470 PRINT
480 INPUT"NUMBER PLAYING";X
490 IF INT(X/2)*2<>X THEN 510
500 GOTO 530
510 PRINT"EVEN NUMBER ONLY..."
```

```
520 PRINT:GOTO 480
530 IF X>10 THEN PRINT:GOTO 480
540 PRINT"NOW ENTER THE";X;"PEOPLE"
550 PRINT"THAT WILL BE PARTNERS,"
560 PRINT"AS EACH WILL PLAY."
570 I=1:M=2
580 INPUT Q$(I)
590 IF I=M THEN 610
600 I=I+1:GOTO 580
610 IF I>=X THEN 630
620 I=I+1:M=M+2:PRINT:GOTO 580
630 REM GET POINT WORD LIST
640 FOR I=1 TO 50:READ W$(I)
650 W(I)=I:NEXT:WM=0
660 I=1
670 CLS:WW$=""
680 PRINT Q$(I);" YOU WILL NOW VIEW THE"
690 PRINT"WORD. PRESS ENTER WHEN"
700 PRINT"READY. YOU'LL HAVE 5 SECONDS"
710 PRINT"TO RECOGNIZE AND REMEMBER IT";
720 INPUT X$
730 REM GET A WORD
740 W=RND(50)
750 IF W(W)=-99 THEN 740
760 W$=W$(W):W(W)=-99:CLS
770 PRINT"THE WORD: ";W$
780 FOR T=1 TO 2000:NEXT:CLS
790 PRINT Q$(I);" GET YOUR PARTNER"
800 PRINT Q$(I+1)," TO THE COMPUTER."
810 PRINT"TIMING WILL BEGIN (OR RESUME)"
820 PRINT"WHEN YOU PRESS ENTER ";Q$(I+1);
830 INPUT X$
840 CLS
850 PRINT"INPUT YOUR WORD THROUGH INKEY$."
860 PRINT"PRESS THE CLEAR KEY AS SOON"
870 PRINT"AS YOU ARE FINISHED, ";Q$(I+1);"."
880 PA=256:PRINT@384,"TIME:";
890 PRINT@393,"SECONDS";:IF TM<>0 THEN 910
900 TM=0:REM TIME RESET
910 PRINT@PA,CHR$(143),
920 V$=INKEY$
930 IF LEN(V$)<>0 THEN 1050
940 IF V$="" THEN 960
950 PRINT@PA,V$;:GOTO 1010
960 FOR TI=1 TO 25:NEXT
970 TM=TM+.2:PRINT@389,INT(TM);
980 PRINT@PA,CHR$(32);
990 FOR TI=1 TO 35:NEXT
```

```
1000 IF INT(TM)=60 THEN 1370 ELSE GOTO 910
1010 REM WORD
1020 WW$=WW$+V$
1030 PA=PA+1
1040 GOTO 970
1050 REM WORD FINISHED
1060 IF ASC(V$)=31 THEN 1080
1070 GOTO 940
1080 PRINT@PA,CHR$(32);
1090 IF WW$<>W$ THEN 1110
1100 GOTO 1160
1110 PRINT@512,"THAT IS NOT THE WORD !!"
1120 FOR TI=1 TO 900:NEXT
1130 PRINT@256,CHR$(30);
1140 PRINT@512,CHR$(30);
1150 WW$="":GOTO 880
1160 PRINT@512,;
1170 PRINT"THAT IS CORRECT, ";Q$(I+1);"!!"
1180 IF WM>=4 THEN 1260
1190 WM=WM+1
1200 PRINT"PLEASE CALL ";Q$(I);" TO THE"
1210 PRINT"COMPUTER FOR ANOTHER "
1220 PRINT"WORD VIEWING, ";Q$(I+1);"."
1230 PRINT Q$(I);" PRESS ENTER";
1240 INPUT X$:WW$=""
1250 GOTO 670
1260 FOR TI=1 TO 2000:NEXT:CLS
1270 PRINT"THAT IS ALL";WM;"WORDS."
1280 PRINT"YOU STILL HAD";60-INT(TM)
1290 PRINT"SECONDS REMAINING."
1300 T(I)=60-INT(TM)
1310 IF I+1>=X THEN 1400 ELSE I=I+2
1320 PRINT"IT IS NOW ";Q$(I);" AND ";
1330 PRINT Q$(I+1);"'S TURN."
1340 PRINT"PRESS ENTER WHEN READY";
1350 INPUT X$
1360 TM=0:WM=0.GOTO 670
1370 PRINT@512,;
1380 PRINT"YOUR TIME HAS EXPIRED !!!"
1390 GOTO 1300
1400 REM ARRANGE TIMES
1410 REM BEST TO WORST
1420 I=1:J=0
1430 IF T(I)>=T(I+2) THEN 1460
1440 T=T(I):T(I)=T(I+2):T(I+2)=T:J=1
1445 V$(I)=Q$(I):V$(I+1)=Q$(I+1)
1450 Q$(I)=Q$(I+2):Q$(I+1)=Q$(I+3)
1455 Q$(I+2)=V$(I):Q$(I+3)=V$(I+1)
```

```
1460 I=I+2
1470 IF I>X THEN 1490
1480 GOTO 1430
1490 IF J=1 THEN 1420
1500 FOR TI=1 TO 2000:NEXT
1510 CLS
1520 PRINT"PLAYERS / TIME CONSUMED"
1530 PRINT"IN ORDER OF TIME "
1540 J=2:FOR I=1 TO X STEP 2
1550 PRINT Q$(I);" & ";Q$(J);
1560 PRINT" CONSUMED";60-T(I);"SECONDS."
1570 FOR TI=1 TO 900:NEXT TI:J=J+2:NEXT I
1580 PRINT
1590 PRINT"END OF RUN . . . "
1600 END
2000 DATA TELEVISION,ACCOMMODATE,TERRACE
2010 DATA STEAM,CONCLUSION,AFFIRMATIVE
2020 DATA SECTION,JOGGING,CIRCLE
2030 DATA GEOMETRIC,CONFUSE,THIRSTY
2040 DATA STOCK,PURGE,VANDALIZE,PERMIT
2050 DATA REDUCE,LIMIT,GLANCE,TARGET
2060 DATA ZODIAC,WIND,WEATHER,FRAGRANCE
2070 DATA MATERAL,SUSPENSE,BLISTER
2080 DATA ALUM,AMBIENT,COMMUNICATE
2090 DATA BYPASS,GULP,FOREIGNER,MORTGAGE
2100 DATA RECOMMEND,PYRAMID,TENDER
2110 DATA RECOVER,BRIDGE,WANDER,WALK
2120 DATA COMPUTER,CORRESPOND,MATH
2130 DATA BLIZZARD,ASHTRAY,HOUSE
2140 DATA AMOUNT,PROFESSION,DEMAND
```

Variables and Strings

X - Number of players, there must be an even number playing
Q$(I) - Players' names
W$(I) - Words
W(I) - For word deleted
WM - Word count
WW$ - Total word entered (V$)
W$ - Current word selected
PA - Print @ location for letter entry
TM - Time loop
V$ - Entry of word through INKEY$
T(I) - Total time consumed by current partners
I - Player count to go on to the next set of partners

Explanation of Program Lines

Lines 10-470 display the general playing instructions.

Lines 480-530 request the number of players. This must be an even number and is tested by line 490.

Lines 540-620 request the players to input their names in pairs.

Lines 630-650 read the words to be guessed from the data lines.

Lines 660-720 display a message.

Lines 730-760 select a word to be guessed. If the word has already been used, it will be deleted by setting W(W)=−99.

Lines 770-780 prints the word so that one of the partners can see it.

Lines 790-830 display a message for the other players (player 2).

Lines 840-1040 accept the input of the word guess by player 2 by using the INKEY$ function to form WW$. Timing is kept by the variable TM which is advanced by +.2. If the integer of TM = 60, one minute has elapsed. If player 2 has entered the word and pressed the clear key (line 1060) the program will branch to line 1080.

Lines 1050-1070 test for the pressing of the clear key.

Lines 1080-1250 determine whether or not the correct word was entered. If it was, another word will be selected, if WM<= 4 in line 1180. If it wasn't, the same word will be given again. In either case continuing depends on the value of INT(TM), the amount of time that has elapsed.

Lines 1260-1300 display a message indicating that all five words have been entered and showing the amount of time remaining.

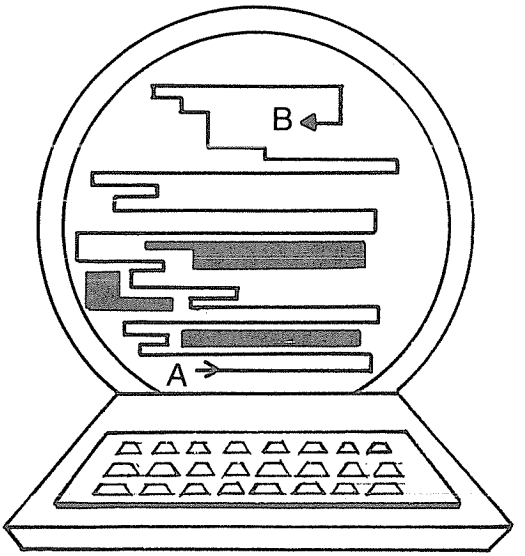
Lines 1310-1360 advance to the next set of partners (if there are more).

Lines 1370-1390 display the time-expired message.

Lines 1400-1490 arrange the times for each set of partners from best to worst.

Lines 1500-1600 display the partners' names and the total amount of time it took them to enter all five words.

Lines 2000-2140 are data lines containing all the words to be guessed.

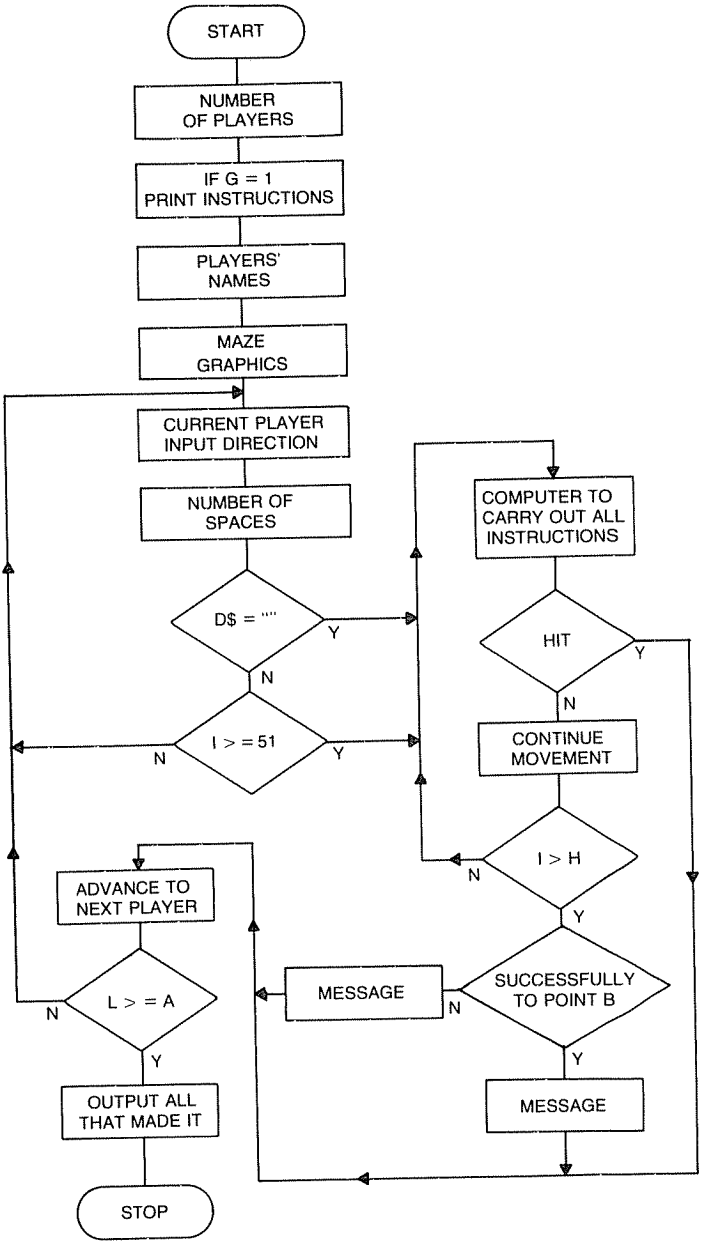


POINT A TO POINT B

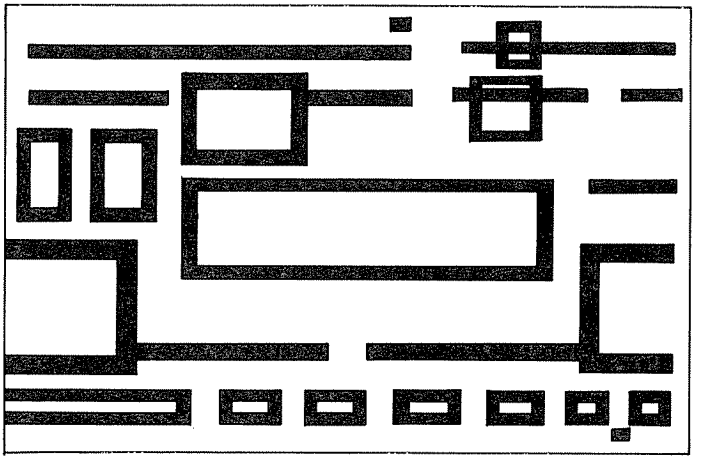
This is one maze game you'll find hard to get away from. Unlike when you use the ordinary maze game (pressing different keys for different directions), you'll enter all of your directions and spaces before any movement takes place. Don't write down all the directions and spaces; you'll be wasting your time! The maze changes with each player or run. Note that the IBM version of this program is included in Appendix A.

Sample Run

```
< FROM POINT A >
< TO POINT B >

INSTRUCTIONS (Y/N)? Y
NUMBER TO PLAY (1-10)? 1

ARE YOU CONFUSED BY THE MANY
DIFFERENT MAZE GAMES THAT ARE
CURRENTLY ON THE MARKET? HAVE
NO FEAR!! FROM POINT A TO
POINT B IS HERE!!
```



Flowchart for *Point A to Point B*.



DIRECTION (L,R,U,D)?—

ACTUALLY, THIS IS NOT YOUR
RUN-OF-THE-MILL MAZE GAME.
WHAT IT IS IS A MAZE GAME THAT
WILL PROBABLY CONFUSE YOU MORE
THAN EVER. YOU SEE, YOU WILL
NOT BE CONSTANTLY BE PRESSING
DIFFERENT KEYS FOR DIFFERENT
DIRECTIONS; YOU'LL INPUT ALL

PRESS ENTER?

OF THE INSTRUCTIONS BEFORE YOU
MOVE AT ALL; UP SO MANY
SPACES, RIGHT SO MANY SPACES,
LEFT SO MANY SPACES, ETC., ETC.
AFTER YOU HAVE ENTERED ALL
YOUR INSTRUCTIONS,
YOU WILL INPUT XX. AT THAT
POINT YOUR PIECE WILL BEGIN
TO MOVE ACCORDING TO THE INSTRUCTIONS
YOU GAVE. IF AT ANY POINT YOU
COLLIDE WITH ANYTHING,
IT IS THE END OF YOUR
TURN...

PRESS ENTER, AGAIN?

```
ENTER THE FIRST NAMES OF
THE 1 PLAYERS, PLEASE
? DAVE

TO BEGIN DAVE, PRESS ENTER
```

Program Listing

```
 10 REM PROGRAM TITLE:
 20 REM FROM POINT A TO POINT B
 30 CLEAR 300:RANDOM:CLS
 40 PRINT"( FROM POINT A )"
 50 PRINT"( TO  POINT  B )"
 60 PRINT:DIM D$(50),S(50)
 70 INPUT"INSTRUCTIONS (Y/N)";W$
 80 IF W$="N" THEN G=1
 90 INPUT"NUMBER TO PLAY (1-10)";A
100 IF A<1 OR A>10 THEN 90
110 IF G=1 THEN G=0:GOTO 410
120 PRINT"ARE YOU CONFUSED BY THE MANY"
130 PRINT"DIFFERENT MAZE GAMES THAT ARE"
140 PRINT"CURRENTLY ON THE MARKET? HAVE"
150 PRINT"NO FEAR!! FROM POINT A TO"
160 PRINT"POINT B IS HERE!!"
170 PRINT"ACTUALLY, THIS IS NOT YOUR"
180 PRINT"RUN-OF-THE-MILL MAZE GAME."
190 PRINT"WHAT IT IS IS A MAZE GAME THAT"
200 PRINT"WILL PROBABLY CONFUSE YOU MORE"
210 PRINT"THAN EVER. YOU SEE, YOU WILL"
220 PRINT"NOT BE CONSTANTLY PRESSING"
230 PRINT"DIFFERENT KEYS FOR DIFFERENT"
240 PRINT"DIRECTIONS; YOU'LL INPUT ALL"
250 INPUT"PRESS ENTER";W
260 CLS
270 PRINT"OF THE INSTRUCTIONS BEFORE YOU"
280 PRINT"MOVE AT ALL: UP SO MANY"
290 PRINT"SPACES, RIGHT SO MANY SPACES,"
300 PRINT"LEFT SO MANY SPACES, ETC.,ETC."
310 PRINT"AFTER YOU HAVE ENTERED ALL"
320 PRINT"OF YOUR INSTRUCTIONS,"
330 PRINT"YOU WILL INPUT XX. AT THAT"
340 PRINT"POINT YOUR PIECE WILL BEGIN"
350 PRINT"TO MOVE ACCORDING TO THE
     INSTRUCTIONS"
360 PRINT"YOU GAVE. IF AT ANY POINT YOU"
370 PRINT"COLLIDE WITH ANYTHING "
380 PRINT"YOUR TURN ENDS.":PRINT
```

```
400 INPUT"PRESS ENTER, AGAIN";W
410 CLS
420 REM PLAYERS
430 REM PLAY IN ORDER OF ENTRY
440 PRINT"ENTER THE FIRST NAMES OF"
450 PRINT"THE";A;"PLAYERS, PLEASE"
460 FOR I=1 TO A:INPUT A$(I)
470 NEXT:L=1:MX=0
480 PRINT
490 PRINT"TO BEGIN ";A$(L);", PRESS ENTER";
500 INPUT W
510 REM GRAPHICS
520 CLS:FOR X=0 TO 127
530 SET(X,0):SET(X,39):NEXT
540 FOR X=1 TO 38
550 SET(0,X):SET(127,X):NEXT:GOTO 580
560 PRINT@65,STRING$(62,140);
565 D=RND(110):IF D<102 THEN 565
570 PRINT@D,STRING$(2,32);:RETURN
580 GOSUB 560:PRINT@193,STRING$(12,131);
590 PRINT@220,STRING$(11,131);:GOTO 620
600 PRINT@232,STRING$(22,131);
605 D=RND(252):IF D<240 THEN 605
610 PRINT@D,STRING$(2,32);:RETURN
620 GOSUB 600:M=30:M1=56:M2=6:M3=12:GOTO 670
630 FOR X=M TO M1:SET(X,M2)
640 SET(X,M3):NEXT:FOR X=M2+1 TO M3-1
650 SET(M,X):SET(M+1,X):SET(M1-1,X)
660 SET(M1,X):NEXT:RETURN
670 GOSUB 630
680 M=1:M1=15:M2=12:M3=17:GOSUB 630
690 M=19:M1=28:GOSUB 630
700 M=M1+3:M1=M1+72:M2=M3-2:M3=M3+6
710 GOSUB 630
720 M=M1+4:M1=M1+27:M2=M2+6:M3=M3+6
730 GOSUB 630
740 M=1:M1=28:GOSUB 630:GOTO 770
750 PRINT@590,STRING$(38,131);
752 PRINT@371,STRING$(12,131);
755 D=RND(626):IF D<590 THEN 755
758 F=RND(381):IF F<371 THEN 758
760 PRINT@D,STRING$(2,32);
765 PRINT@F,STRING$(2,32);:RETURN
770 GOSUB 750:M1=M1-4:M2=M3+2:M3=M3+6
775 GOSUB 630
780 M=M1+4:M1=M1+14
790 GOSUB 630
800 IF M>=112 THEN 880
```

```
810 GOTO 780
820 M=M-15:M1=M+10:M2=2:M3=5
830 GOSUB 630
840 M=M-14:M1=M+12:M2=M3+3:M3=M3+8
850 GOSUB 630
860 M=M-15:M1=M+4:M2=4:M3=6
870 GOSUB 630:RETURN
880 REM START / STOP LOCATIONS
890 GOSUB 820:XX=1:Y=37:X=2:X2=125
900 X1=RND(X2)
910 RESET(X-1,Y)
920 X=X+1
930 SET(X,Y)
940 XX=XX+1:IF G=1 AND XX)=150 THEN 1000
950 IF X=X1 AND XX)=50 THEN 1000
960 IF X)=X2 THEN 980
970 GOTO 900
980 RESET(X,Y):RESET(X-1,Y)
990 X=2:GOTO 900
1000 XX=1:PRINT@896,;:IF G=1 THEN 1110
1010 PRINT"THIS IS POINT A (START)."
1020 RESET(X,Y):RESET(X-1,Y)
1030 FOR T=1 TO 50:NEXT
1040 SET(X,Y)
1050 FOR T=1 TO 25:NEXT:XX=XX+1
1060 IF XX(=10 THEN 1020
1070 IF G=1 THEN 1150 ELSE JJ=X
1080 G=1:PRINT@896,CHR$(30)
1090 PRINT@896,"LOOK AT THE TOP (POINT B)."
1100 X=2:Y=2:XX=1:X1=1:X2=90:GOTO 900
1110 PRINT@896,CHR$(30)
1120 PRINT@896,;
1130 PRINT"THIS IS POINT B (STOP ZONE)."
1140 R=X:GOTO 1020
1150 PRINT@896,CHR$(30)
1160 GOTO 1210
1170 PRINT@896,;
1180 RETURN
1190 FOR T=1 TO 1200:NEXT
1200 PRINT@896,CHR$(30):RETURN
1210 GOSUB 1170
1220 PRINT"YOU WILL NOW GIVE INSTRUCTIONS"
1230 GOSUB 1190:GOSUB 1170
1240 PRINT"TO ENTER THE STOP ZONE..."
1250 GOSUB 1190:GOSUB 1170
1260 PRINT"THE STOP ZONE BLOCK MUST BE HIT"
1270 GOSUB 1190:GOSUB 1170
1280 PRINT"HEAD-ON TO COMPLETE A RUN."
```

```
1290 GOSUB 1190:GOSUB 1170
1300 I=1
1310 INPUT"DIRECTION (L,R,U,D)";D$
1320 IF D$="XX" THEN 1380
1330 D$(I)=D$:GOSUB 1170
1340 INPUT"FOR HOW MANY SPACES";S
1350 S(I)=S:I=I+1:GOSUB 1170
1360 IF I>=51 THEN 1430
1370 GOSUB 1200:GOSUB 1170:GOTO 1310
1380 GOSUB 1170
1390 PRINT"THE COMPUTER WILL NOW CARRY"
1400 GOSUB 1190:GOSUB 1170
1410 PRINT"OUT ALL OF YOUR INSTRUCTIONS."
1420 I=I-1:H=I:I=1:X=JJ:Y=37:G=0
1425 GOSUB 1190:GOTO 1480
1430 GOSUB 1170
1440 PRINT"THAT IS THE LIMIT OF THE"
1450 GOSUB 1190:GOSUB 1170
1460 PRINT"INSTRUCTIONS YOU CAN ENTER."
1470 GOSUB 1190:GOTO 1380
1480 D$=D$(I):S=S(I):J=1
1490 IF D$="L" THEN 1540
1500 IF D$="R" THEN 1580
1510 IF D$="U" THEN 1620
1520 IF D$="D" THEN 1660
1530 GOTO 1760:REM BAD DIRECT
1540 IF POINT(X-1,Y) THEN 1760
1550 X=X-1:RESET(X+1,Y)
1560 IF X<=0 THEN 1760
1570 GOTO 1690
1580 IF POINT(X+1,Y) THEN 1760
1590 X=X+1:RESET(X-1,Y)
1600 IF X>=126 THEN 1760
1610 GOTO 1690
1620 IF POINT(X,Y-1) THEN 1760
1630 Y=Y-1:RESET(X,Y+1)
1640 IF Y<=1 THEN 1760
1650 GOTO 1690
1660 IF POINT(X,Y+1) THEN 1760
1670 Y=Y+1:RESET(X,Y-1)
1680 IF Y>=39 THEN 1760
1690 SET(X,Y)
1700 IF S>J THEN 1720
1710 GOTO 1730
1720 J=J+1:GOTO 1490
1730 I=I+1
1740 IF I>H THEN HH=1:GOTO 1760
1750 GOTO 1480
```

```
1760 GOSUB 1170:IF ABS(X-R)<=1 THEN 2060
1765 IF HH=1 THEN HH=0:GOTO 1900
1770 PRINT"SORRY...YOU HAVE ATTEMPTED"
1780 GOSUB 1190:GOSUB 1170
1790 PRINT"A SHORT-CUT THAT DIDN'T WORK."
1800 REM ANOTHER PLAY?
1810 GOSUB 1190
1820 GOTO 1950
1830 GOSUB 1190:GOSUB 1170
1840 PRINT"READY TO TRY AGAIN (Y/N)";
1850 INPUT W$
1860 IF W$="Y" THEN 1880
1870 GOTO 2300
1880 RESET(X,Y):RESET(R,2)
1890 GOTO 1970
1900 GOSUB 1170
1910 PRINT"THAT WASN'T ENOUGH TO GET"
1920 GOSUB 1190:GOSUB 1170
1930 PRINT"YOU TO THE STOP ZONE ";A$(L)
1940 GOTO 1800
1950 REM PLAYER ADVANCE
1960 IF L>=A THEN 2020 ELSE L=L+1
1970 GOSUB 1170
1980 PRINT A$(L);" WILL NOW ATTEMPT THE"
1990 GOSUB 1190:GOSUB 1170
2000 PRINT"VENTURE FROM POINT A TO B."
2005 RESET(X,Y):RESET(R,2):GOSUB 560
2010 GOSUB 600:GOSUB 750:M=112:M1=122
2015 GOSUB 1190:GOTO 880
2020 REM START OVER
2030 L=1:GOSUB 1170
2040 PRINT"ALL HAVE PLAYED...."
2050 GOSUB 2160:MX=0:GOTO 1830
2060 IF ABS(Y-2)>=1 THEN 1900
2070 GOSUB 1190:GOSUB 1170:REM COMPLETE
2080 PRINT"VERY WELL DONE, ";A$(L);"!!"
2090 GOSUB 1190:GOSUB 1170
2100 PRINT"YOU HAVE MADE A COMPLETE"
2110 GOSUB 1190:GOSUB 1170
2120 PRINT"PASSAGE TO POINT B...."
2130 GOSUB 1190
2140 M(L)=1:MX=1
2150 GOTO 1950
2160 REM PLAYERS MADE TO POINT B
2170 IF MX=0 THEN RETURN
2180 GOSUB 1190:GOSUB 1170
2190 PRINT"THE FOLLOWING HAVE EASILY MADE"
2200 GOSUB 1190:GOSUB 1170
```

```
2210 PRINT"IT TO POINT B....."
2220 GOSUB 1190:GOSUB 1170
2230 FOR U=1 TO A
2240 IF M(U)=1 THEN 2260
2250 NEXT:RETURN
2260 PRINT A$(U);"....."
2270 GOSUB 1190:GOSUB 1170
2280 M(U)=0:GOTO 2250
2300 REM END
2310 GOSUB 1190:GOSUB 1170
2320 PRINT"POINT A TO POINT B"
2330 GOSUB 1190:GOSUB 1170
2340 PRINT"TERMINATED....."
2350 GOSUB 1190:GOSUB 1170
2360 END
```

Variables and Strings

A - Number of players
G - 0/1 for skip of the instructions
A$(I) - Player's first names
L - Current player
D,F - A random amount to make certain parts of each maze different
M,M1,M2,M3 - For maze graphics
X,Y - Start location; JJ=X at point A
X,Y - Stop location; R=X at point B
D$(I) - User's direction
S(I) - Spaces per direction
I - Number of moves, up to 50
H - Total number of moves entered
D$ - Direction, as per D$(I)
S - Spaces, as per S(I)
J - Total spaces per direction (counter)
M(L) - Players that made it to point B

Explanation of the Program Lines

Lines 10-400 request the number of players and display the general instructions.

Lines 410-470 accept the players' first names.

Lines 480-500 display a message to the current player.

Lines 510-870 create the maze graphics. The random variable D and F will help insure that certain parts of the maze will be different each time in case there are users who decide to cheat by writing down the instructions.

Lines 880-1160 randomly select the locations of point A, the start, and point B, the stop.

Lines 1170-1180 form a subroutine used for displaying messages.

Lines 1190-1200 form a subroutine that creates a pause and clears the message from the screen.

Lines 1210-1290 display a message to the user before he or she inputs his or her instructions.

Lines 1300-1370 allow the user to input all of his or her instructions for getting to point B. Up to a total of 50 can be entered if they are needed.

Lines 1380-1410 display a message indicating that the computer is ready to carry out the user's instructions.

Lines 1420-1425 set some variables before the actual movement begins.

Lines 1430-1470 displays a message indicating that no more instructions can be entered.

Lines 1480-1750 cause the computer to carry out all of the instructions entered by the user. These lines also test for any hits. If a hit is encountered, the program is terminated or branches to the next player.

Line 1760 tests to see if the user is close to point B.

Line 1765 tests to see if the user has entered enough instructions.

Lines 1770-1820 display a message if a hit is encountered.

Lines 1830-1890 ask if the user wants to try again after all have played.

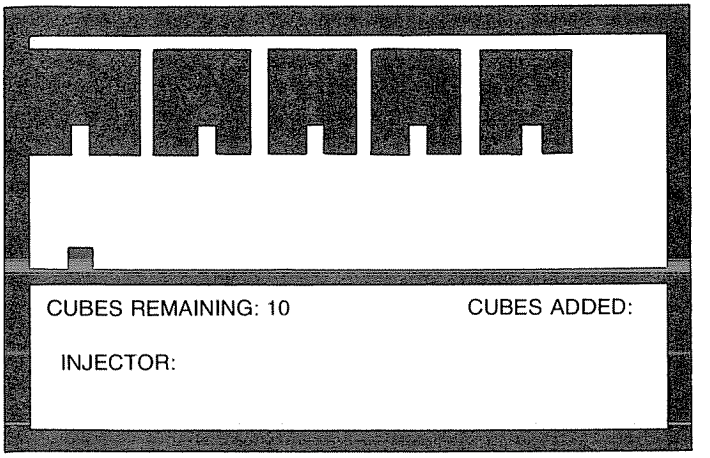
Lines 1900-1940 display a message indicating that there are not enough instructions to make it to point B.

Lines 1950-2015 advance to the next player.

Lines 2020-2050 display a message indicating that all have played and allow the user to try again!

Line 2060 further tests to see if the player is at point B. If he is not, the program branches back to line 1900.

Lines 2070-2150 print the names of all players that have successfully made it to point B and then terminate the program.

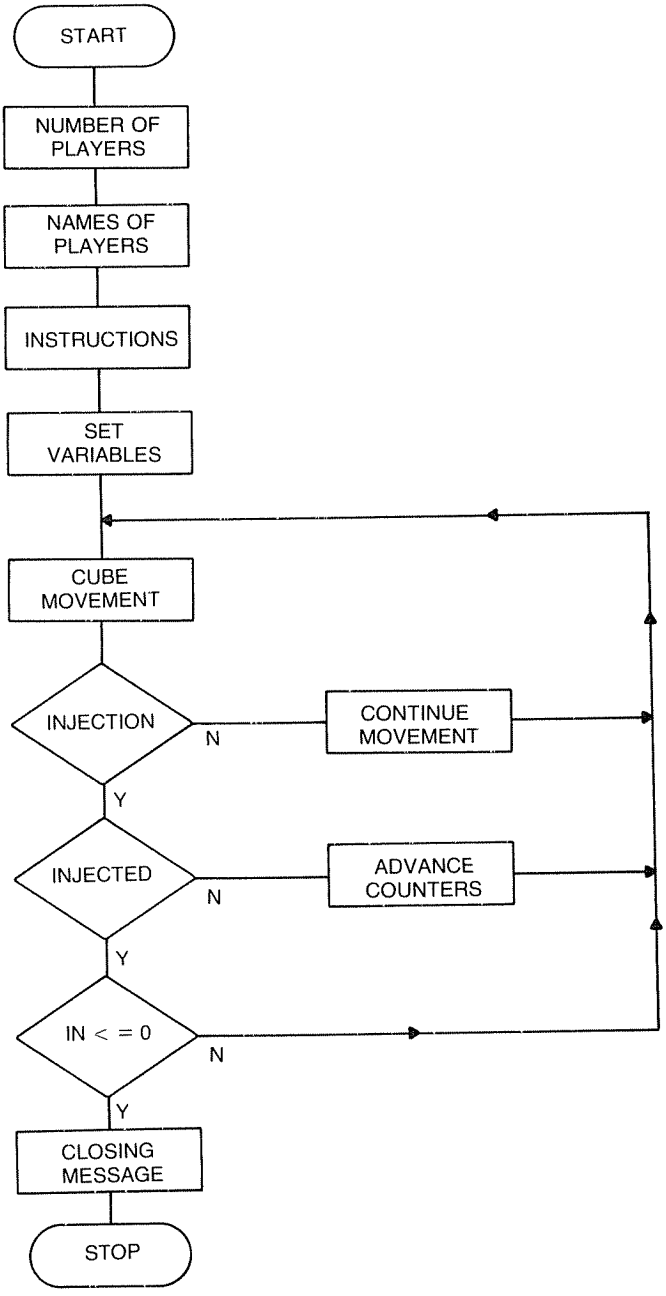


INJECTION

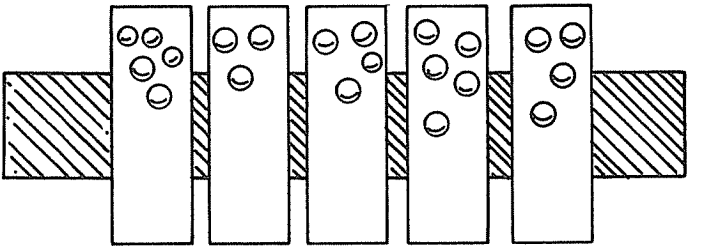
In a laboratory there is a colony of bacteria. This colony was meant for experiments, but now it cannot be controlled. You'll have the honor of trying to destroy it. As many as five cubes of bacteria will race across the video one at a time. Your injector machine is located on the runner near the middle of the video. To inject, press the up arrow key. For an injection to work, it must hit directly in the opening of the bottom of the cube. If an injection is missed, the bacteria increase by one. To be successful, destroy all the cubes. Refer to pages 1-3 for information on the sound routine. Note that the IBM version of this program is included in Appendix A, and the Apple version in Appendix B.

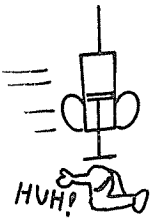
Sample Run

```
> INJECTION <

NUMBER TO PLAY? 4
THE INITIALS OF PLAYERS.
? A.A.
? B.B.
? C.C.
? D.D.

YOU ARE HEAD OF A LABORATORY
IN WHICH YOUR COMPANY IS
INJECTING CUBES WITH
```



Flowchart for *Injection*.





A SOLVENT TO DESTROY A COLONY OF BACTERIA THAT CANNOT BE CONTROLLED. THESE CUBES WILL RACE ACROSS THE SCREEN 5 AT A TIME. TO INJECT A CUBE, HOLD THE UP ARROW KEY. THE SOLVENT MUST GO IN THE OPENING AT THE BOTTOM OF THE CUBES.

PRESS ENTER?

EACH TIME A CUBE IS INJECTED, THE NUMBER RACING ACROSS THE VIDEO WILL DECREASE BY ONE, BUT NOT LESS THAN THREE WILL RACE ACROSS AT ONE TIME. A TOTAL OF 10 CUBES MUST BE INJECTED. IF AT ANYTIME YOU MISS AN INJECTION, THE NUMBER WILL INCREASE BY ONE. THE INJECTION MACHINE CAN BE MOVED LEFT OR RIGHT WITH THE LEFT AND RIGHT ARROW KEYS. GOOD LUCK...

PRESS ENTER TO BEGIN?

Program Listing

```
 10 REM PROGRAM TITLE: INJECTION
 20 CLEAR 100
 30 RANDOM
 40 CLS
 50 PRINT"> INJECTION <"
 60 PRINT:S=31266
 70 INPUT"NUMBER TO PLAY (1-10)";U
 80 IF U<1 OR U>10 THEN 70
 90 PRINT"THE INITIALS OF PLAYERS."
100 FOR I=1 TO U:INPUT K$(I)
110 NEXT:PRINT:PL=1
120 PRINT"YOU ARE HEAD OF A LABORATORY"
130 PRINT"IN WHICH YOUR COMPANY IS "
140 PRINT"INJECTING CUBES WITH"
150 PRINT"A SOLVENT TO DESTROY A COLONY"
160 PRINT"OF BACTERIA THAT CANNOT BE"
170 PRINT"CONTROLLED. THESE CUBES WILL"
180 PRINT"RACE ACROSS THE SCREEN 5 AT A"
190 PRINT"TIME. TO INJECT A CUBE, HOLD"
200 PRINT"THE UP ARROW KEY. THE INJECTED"
210 PRINT"SOLVENT MUST GO IN THE OPENING"
220 PRINT"AT THE BOTTOM OF THE CUBES."
230 PRINT
240 PRINT
250 INPUT"PRESS ENTER";L
260 PRINT@64,CHR$(30)
270 PRINT"EACH TIME A CUBE IS INJECTED,"
280 PRINT"THE NUMBER OF CUBES RACING ACROSS"
290 PRINT"THE VIDEO WILL DECREASE BY ONE, BUT"
300 PRINT"NOT LESS THAN THREE WILL RACE"
310 PRINT"ACROSS AT ONE TIME. A TOTAL"
320 PRINT"OF 10 CUBES MUST BE INJECTED."
330 PRINT"IF AT ANYTIME YOU MISS AN"
340 PRINT"INJECTION, THE NUMBER WILL"
350 PRINT"INCREASE BY ONE. THE INJECTION"
360 PRINT"MACHINE CAN BE MOVED LEFT OR"
370 PRINT"RIGHT, WITH THE LEFT AND RIGHT"
380 PRINT"ARROW KEYS. GOOD LUCK..."
390 PRINT
400 INPUT"PRESS ENTER TO BEGIN";L
410 CLS:JN=0:T=1:IN=10:E=0:CV=5:GOSUB 600
420 REM THE CUBES
430 X=65:Y=129:M=X:N=Y:J=0:JJ=0
440 PRINT@X,STRING$(6,191);
450 PRINT@Y,STRING$(6,191),
460 PRINT@Y+2,"  ";
```

```
470 X=X+8.Y=Y+8:J=J+1:POKE S,25
480 IF JJ=1 THEN 520
490 IF J>=CV THEN 510
500 GOTO 440
510 REM BLANK OUT
520 PRINT@M,STRING$(6,32);
530 PRINT@N,STRING$(6,32);
540 M=M+8:N=N+8:SS=USR(0)
550 REM RECYCLE
560 IF X>=128 THEN X=65:Y=129:JJ=1:GOTO 580
570 IF M>=128 THEN 430
580 GOTO 720
600 REM OUTER BORDER / INJECTOR
610 FOR X=0 TO 127:SET(X,0)
620 SET(X,46):SET(X,47):NEXT
630 FOR X=1 TO 45
640 SET(0,X):SET(1,X)
650 SET(126,X):SET(127,X):NEXT
660 PRINT@576,STRING$(63,191);
670 Q=514:I$=STRING$(2,191):CA=0
680 PRINT@Q,I$;:PRINT@770,"INJECTOR:";
690 PRINT@642,"CUBES REMAINING:";
700 PRINT@682,"CUBES ADDED:";
710 PRINT@780,K$(PL);:RETURN
720 REM INJECTION / INJECTOR MOVE
730 IF R=8 THEN 790
740 R=PEEK(14440)
750 IF R=8 THEN 790
760 IF R=32 THEN 900
770 IF R=64 THEN 960
780 GOTO 440
790 IF JN=1 THEN 810
800 QQ=Q:JN=1
810 IF Q<=191 THEN V=1:GOTO 860
820 PRINT@Q,"  ";
830 Q=Q-384:PRINT@Q,I$;
835 IF PEEK(Q+15360+2)=191 THEN 845
840 GOTO 440
845 IF PEEK(Q+15360-2)=191 THEN 1020
850 GOTO 440
860 PRINT@Q,"  ";
870 Q=QQ+E:QQ=0:E=0
880 PRINT@Q,I$;:IF V=1 GOSUB 1140
890 R=0:JN=0:GOTO 440
900 REM TO LEFT
910 IF Q<=516 THEN 440
920 PRINT@Q,"  ";
930 Q=Q-T
```

```
 940 PRINT@Q,I$;
 950 GOTO 440
 960 REM TO RIGHT
 970 IF Q>=570 THEN 440
 980 PRINT@Q,"  ";
 990 Q=Q+T
1000 PRINT@Q,I$;
1010 GOTO 440
1020 IF PEEK(Q+15360-64)=191 THEN 1030
1025 GOTO 440
1030 IF PEEK(Q+15360-1)=191 THEN 1045
1040 GOTO 440
1045 IF PEEK(Q+15360+1)=191 THEN 1060
1050 GOTO 440
1060 REM INJECTED
1070 T=T+1:CV=CV-1:POKE S,255:SS=USR(0)
1080 IF T>=5 THEN T=1
1090 IF CV<=2 THEN CV=5
1100 IN=IN-1:PRINT@660,IN;
1110 IF IN<=0 THEN 1250
1120 E=RND(5):IF E+QQ>=572 THEN E=-RND(5)
1130 GOTO 860
1140 REM CUBE ADDED / INJECTION MISSED
1150 IN=IN+1:POKE S,50:SS=USR(0)
1160 CA=CA+1:V=0
1170 T=T-1
1180 IF T<=0 THEN T=1
1190 PRINT@660,IN;
1200 PRINT@695,CA;
1210 RETURN
1250 REM PLAYER ADVANCE
1260 W(PL)=CA:PL=PL+1
1270 IF PL>U THEN 1310
1280 PRINT@770,"STANDBY ";K$(PL);".....";
1290 FOR YY=1 TO 1500:NEXT
1300 R=0:GOTO 410
1310 FOR YY=1 TO 900:NEXT:CLS
1320 PRINT"ALL PLAYERS HAVE HAD THEIR"
1330 PRINT"TURN AT THE INJECTION MACHINE."
1340 PRINT"SCORES ARE BASED ON THE NUMBER"
1350 PRINT"OF CUBES THAT WERE ADDED DURING"
1360 PRINT"THE INJECTION PROCESS."
1370 PRINT:FOR I=1 TO U
1380 PRINT"INJECTOR: ";K$(I)
1390 PRINT"HAD";W(I);"CUBES ADDED."
1400 PRINT"FINAL SCORE EVALUATION: ";
1410 W=W(I)
1420 IF W>=0 AND W<=5 THEN 1470
```

```
1430 IF W>=5 AND W<=10 THEN 1480
1440 IF W>=10 AND W<=20 THEN 1490
1450 PRINT"TO SORRY TO MENTION."
1460 GOTO 1500
1470 PRINT"EXCELLENT.":GOTO 1500
1480 PRINT"FAIR.":GOTO 1500
1490 PRINT"POOR."
1500 FOR YY=1 TO 1200:NEXT YY:PRINT
1510 NEXT I
1520 PRINT
1530 PRINT"END OF INJECTION."
1540 END
```

Variables and Strings

S - Sound location (machine language)
U - Number of players 1-10
K$(I) - Initials of players
PL - Total number of players, counter 1 to U
JN - 0/1 variable; 1 means that QQ is already set to the amount of Q (before injection)
IN - Total injections
E - Random amount added to Q when it returns to its former location (after injection)
X,Y,M,N,J,JJ - All for graphics of cubes
SS - USR call, sound routine
Q - Print @ location of injector
I$ - Injector
CA - Cubes added, injection missed
R - Peek for users key closure
QQ - Value of Q (JN=1) before injection (Q location)
T - Amount added to Q if cube is injected; this amount will not be larger than 4
W(PL) - Player, number of cubes injected
W - Value of W(PL) for print messages

Explanation of the Program Lines

Lines 10-400 accept the input of the number of players and their names, and display the general instructions.

Line 410 initializes some variables, and uses lines 600-710 to set the outer border and print a message.

Lines 420-570 draw the cubes (5) and blank out a portion at the lower center of each. Lines 440-580 also keep the cubes moving.

Lines 720-770 test for key closure and determine whether or not R is equal to 8.

Line 780 recycles back to line 440 to keep the cubes moving.

Line 790 tests whether or not JN=1. If it does QQ already contains the value of Q. This is the value before the injection.

Line 800 sets the variable QQ to the value of Q, meaning that the user is ready to make an injection. JN is set to 1.

Line 810 tests the value of Q for resetting purposes. Lines 860-890 are used.

Lines 820-850 tests to see if a cube has been injected. If one has been, the program will branch to the appropriate lines.

Lines 900-950 allow the user to move the injector to the left.

Lines 960-1010 allow the user to move the injector to the right.

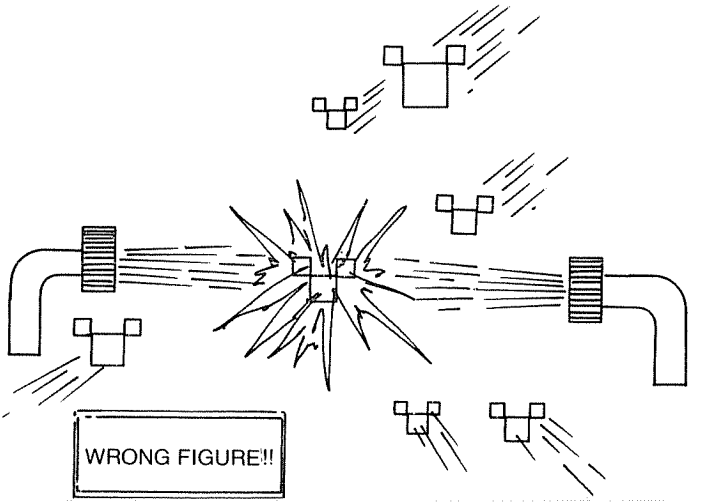
Lines 1020-1050 test peek(Q+15360 + or – an amount) to see if the cube has really been injected.

Lines 1060-1130 are used when the cube had been injected. The variable IN is decreased by one. A random amount E, is added to Q so the user cannot expect another injection to occur in the same location at that moment.

Lines 1140-1210 are used when the injection missed. Another cube is added. The total number of injections and cubes are printed in lines 1190 and 1200.

Lines 1250-1310 go to the next player if PL (player counter) is less than U (total players). If this argument is satisfied, the program branches back to line 410. The variables are reset and the next player begins.

Lines 1320-1540 terminate the program. All players' names, their scores, and messages are printed.

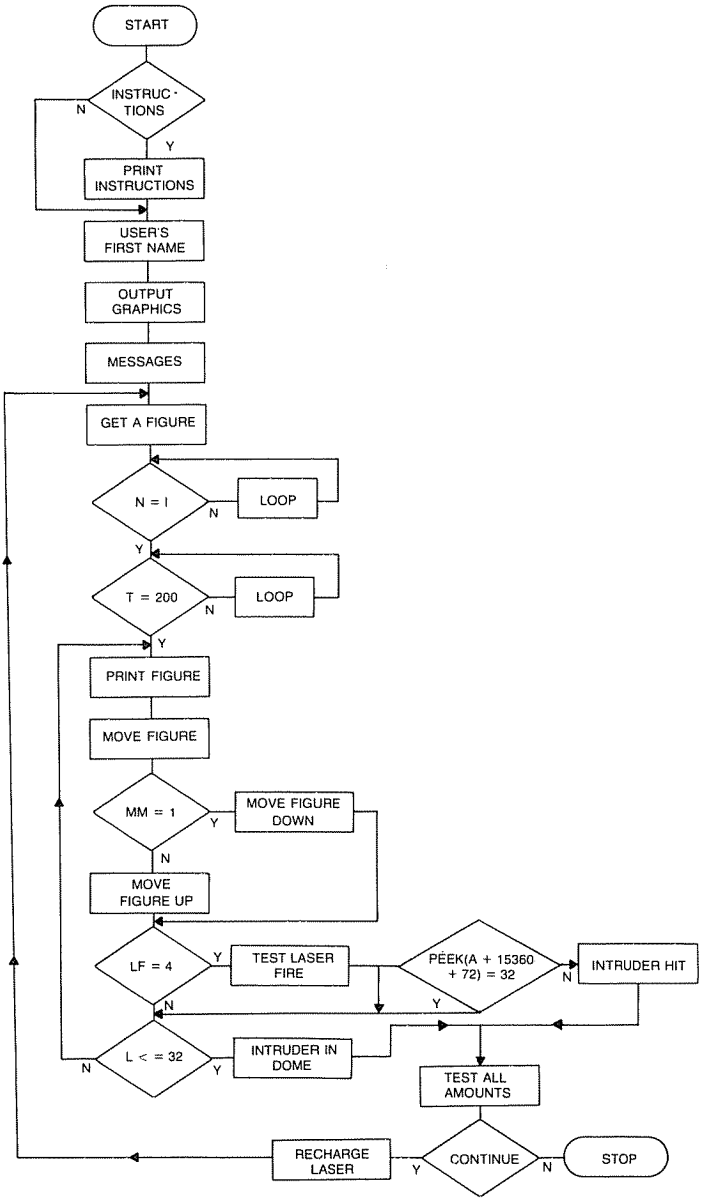


INTERCEPT

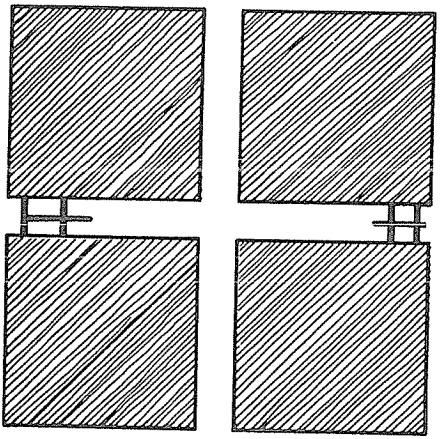
This game requires you to intercept and possibly destroy any intruder that wanders into your dome. You have been given a ten day commission to guard this city within a dome. Some of the figures that enter the dome are alien; some are friendly. Within an allotted amount of time, you must decide which is which. It helps to remember the shape of the figures. Do not destroy the friendly intruders! Note that the IBM version of this program is included in Appendix A.

Sample Run

```
INTERCEPT

INSTRUCTIONS? YES
YOU ARE THE LEADER OF A SPECIAL
GROUP KNOWN AS INTERCEPT.
YOU WILL HAVE TO IDENTIFY AND
POSSIBLY DESTROY ANY ALIENS THAT
VENTURE IN THIS STERILIZED DOME
WHERE ALL OF YOUR INHABITANTS
LIVE. THE DOME, WHICH CONTAINS
CROSSROADS, IS THE TERRITORY.
NEAR THE EDGES OF THE SCREEN
```



Flowchart for *Intercept*.



LEADER:
PRESENT CONDITION: QUIET
LASER STATUS: CHARGING
DAYS ON COMMISSION: 10

ARE TWO LASER CANNONS.
THESE CANNONS WILL FIRE ONLY
WHEN THE CORRECT KEY SEQUENCE
HAS BEEN ENTERED BY YOU.

PRESS ENTER PLEASE?

AT RANDOM TIMES, DIFFERENT FIGURES
WILL APPEAR AT THE CENTER OF
THE VERTICAL ROAD OF THE CROSS-
ROADS. WHEN THE FIGURE IS AT THE
CENTER IN LINE WITH YOUR LASER,
YOU MUST DECIDE WHETHER IT IS ALIEN
OR FRIENDLY. IF YOU BELIEVE THAT
IT IS THE LATTER OF THE TWO, DO
NOTHING; JUST LET IT PASS. IF
YOU BELIEVE IT IS ALIEN, FIRE THE
LASER BY PRESSING THE RIGHT THEN
THE LEFT ARROW KEYS, THEN FINALLY
PRESSING THE SPACEBAR. YOU CAN
HAVE THE LASER READY BEFOREHAND
BY PRESSING THE ARROW KEYS TO

PRESS ENTER PLEASE?

MAKE THE LASER READY, YOUR COM-
MISSION WILL LAST 10 DAYS. IF
YOU DESTROY 10 FRIENDLY IN-
TRUDERS BEFORE THAT TIME, YOUR
COMMISSION WILL BE TERMINATED
BEFORE THEN.
THE FINAL OBJECT IS TO DESTROY AS
MANY ALIEN INTRUDERS AS YOU CAN
BEFORE YOUR COMMISSION EXPIRES.

YOUR LAST NAME, PLEASE? SMITH

Program Listing

```
 10 REM PROGRAM TITLE: INTERCEPT
 20 CLEAR 200:RANDOM
 30 CLS:SS=31266
 40 PRINT CHR$(157);" INTERCEPT ";CHR$(174)
 50 PRINT
 60 INPUT"INSTRUCTIONS (Y/N)";Q$
 70 IF Q$="N" OR Q$="NO" THEN 520
 80 PRINT
 90 PRINT"YOU ARE THE LEADER OF A SPECIAL"
100 PRINT"GROUP KNOWN AS INTERCEPT."
110 PRINT"YOU WILL HAVE TO IDENTIFY AND"
120 PRINT"POSSIBLY DESTROY ANY ALIENS THAT"
130 PRINT"VENTURE IN THE STERILIZED DOME"
140 PRINT"WHERE ALL OF YOUR INHABITANTS"
150 PRINT"LIVE. THE DOME IS SQUARE AND"
160 PRINT"HAS A CROSSROAD WITHIN IT."
170 PRINT"AT THE LEFT AND RIGHT EDGE OF THE
    HORIZONTAL"
180 PRINT"ROAD ARE TWO LASER CANNONS."
190 PRINT"THESE CANNONS WILL FIRE ONLY"
200 PRINT"WHEN THE CORRECT KEY SEQUENCE"
210 PRINT"HAS BEEN ENTERED BY YOU."
220 PRINT
230 INPUT"PRESS ENTER PLEASE";Q$
240 CLS
250 PRINT"AT RANDOM TIMES,DIFFERENT FIGURES"
260 PRINT"WILL APPEAR AT THE CENTER OF"
270 PRINT"THE VERTICAL ROAD OF THE CROSS-"
280 PRINT"ROADS. WHEN THE FIGURE IS AT THE"
290 PRINT"CENTER IN LINE WITH YOUR LASERS"
```

```
300 PRINT"YOU MUST DECIDE WHETHER IT IS ALIEN"
310 PRINT"OR FRIENDLY. IF YOU BELIEVE THAT"
320 PRINT"IT IS THE LATTER OF THE TWO, DO"
330 PRINT"NOTHING; JUST LET IT PASS. IF"
340 PRINT"YOU BELIEVE IT IS ALIEN,FIRE THE"
350 PRINT"LASER BY PRESSING THE RIGHT THEN"
360 PRINT"THE LEFT ARROW KEYS, AND THEN
    FINALLY"
370 PRINT"PRESSING THE SPACEBAR. YOU CAN"
380 PRINT"HAVE THE LASER READY BEFOREHAND"
390 PRINT"BY PRESSING THE ARROW KEYS TO"
400 INPUT"PRESS ENTER PLEASE";Q$
410 CLS
420 PRINT"MAKE THE LASER READY. YOUR COM-"
430 PRINT"MISSION WILL LAST 10 DAYS. IF"
440 PRINT"YOU DESTROY 10 FRIENDLY IN-"
450 PRINT"TRUDERS BEFORE THAT TIME, YOUR"
460 PRINT"COMMISSION WILL BE TERMINATED"
470 PRINT"BEFORE THEN."
480 PRINT"THE FINAL OBJECT IS TO DESTROY AS"
490 PRINT"MANY ALIEN INTRUDERS AS YOU CAN"
500 PRINT"BEFORE YOUR COMMISSION EXPIRES."
510 PRINT
520 PRINT
530 INPUT"YOUR LAST NAME, PLEASE";P$
540 CLS
550 REM INTERCEPT CROSSROADS
560 X$=STRING$(26,191)
570 A=5:J=1
580 PRINT@A,X$:PRINT@A+320,X$;
590 PRINT@A+30,X$;:PRINT@A+350,X$;
600 A=A+64:J=J+1
610 IF J>=5 THEN 630
620 GOTO 580
630 AA=315:A$=CHR$(157)+CHR$(183)
640 AA$=CHR$(187)+CHR$(174)
650 PRINT@A,A$;:PRINT@AA,AA$;
660 PRINT@581,"LEADER: ";P$
670 PRINT@645,"PRESENT CONDITION:"
680 PRINT@709,"LASER STATUS:"
690 PRINT@773,"DAYS TO GO ON COMMISSION:"
700 REM GET FIGURES
710 REM FRIENDLY
720 F$(1)=CHR$(134):F$(2)=CHR$(139)
730 F$(3)=CHR$(141):F$(4)=CHR$(153)
740 F$(5)=CHR$(166)
750 REM ALIEN
760 A$(1)=CHR$(135):A$(2)=CHR$(137)
```

```
 770 A$(3)=CHR$(142):A$(4)=CHR$(158)
 780 A$(5)=CHR$(173)
 790 REM DOUBLE SIZE
 800 FOR I=1 TO 5
 810 F$(I)=F$(I)+F$(I)
 820 A$(I)=A$(I)+A$(I)
 830 NEXT:K1=A+2:K2=A+28
 840 REM DAYS / ALIEN / FRIENDLY
 850 DA=10:AI=0:FI=0:AD=0:FD=0:DF=0
 860 C$="QUIET    ":CC$="ALERT"
 870 L$="CHARGING":LL$="READY    "
 880 PRINT@667,C$;
 890 PRINT@731,L$;
 900 PRINT@798,DA;
 910 REM START LOCATION
 920 L=544
 930 REM CONDITION
 940 FOR T=1 TO 500
 950 I=RND(5)
 960 N=RND(5)
 970 IF N=I THEN 1000
 980 NEXT T
 990 GOTO 940
1000 REM ALERT / INTRUDER
1005 IF T<=200 THEN T=T+1:GOTO 1005
1010 X=0:LF=1:XC=0:DS=0:YU=0
1020 PRINT@667,STRING$(6,32);
1030 FOR T=1 TO 100:NEXT
1040 PRINT@667,CC$;:POKE SS,100:S=USR(0)
1050 FOR TT=1 TO 150:NEXT
1060 X=X+1:IF X<=4 THEN 1020
1070 PRINT@667,"INTRUDER";
1080 R=RND(2)
1090 IF R=2 THEN 1110
1100 PRINT@L,F$(N);:GOTO 1115
1110 PRINT@L,A$(N);
1115 YU=YU+1
1120 IF E=0 THEN E=PEEK(15340)
1125 IF LF=4 THEN 1360
1130 MM=RND(2)
1140 IF MM=1 AND YU<=70 THEN 1180
1150 IF L<=32 THEN 1520
1160 L=L-64:PRINT@L+64,STRING$(2,32);
1170 IF LF<=3 GOTO 1200 ELSE 1360
1180 IF L>=544 THEN MM=2:GOTO 1150
1190 L=L+64:PRINT@L-64,STRING$(2,32);
1200 IF E=32 AND LF=1 THEN 1240
1210 IF E=64 AND LF=2 THEN 1250
```

```
1220 IF E=128 AND LF=3 THEN 1260
1230 GOTO 1270
1240 LF=2:GOTO 1280
1250 LF=3:GOTO 1280
1260 LF=4:GOTO 1280
1270 E=0:GOTO 1090
1280 PRINT@837,"LASER KEY SEQUENCE:";
1290 PRINT LF-1:POKE SS,30:S=USR(0)
1300 IF LF=3 THEN 1320
1310 GOTO 1270
1320 PRINT@731,LL$;
1330 GOTO 1270
1360 REM FIRING
1370 IF R=1 THEN 1390
1380 PRINT@L,A$(N);:GOTO 1400
1390 PRINT@L,F$(N);
1400 PRINT@A+2,STRING$(25,140);
1410 PRINT@AA-25,STRING$(25,140);
1415 FOR TT=1 TO 30:POKE SS,15:S=USR(0)
1417 NEXT
1420 IF PEEK(A+15360+27)=32 THEN 1440
1430 GOTO 1650
1440 PRINT@731,L$;:LF=1
1450 PRINT@837,"INDIRECT HIT          ";
1460 PRINT@K1,STRING$(25,32);
1470 PRINT@K2,STRING$(26,32);
1480 FOR TT=1 TO 20:POKE SS,45
1490 S=USR(0):NEXT
1500 GOTO 1090
1520 REM INTRUDER IN DOME
1525 IF YU<=50 THEN 1190
1530 PRINT@L,"  ";
1540 PRINT@837,"INTRUDER IS NOW IN DOME..."
1550 FOR TT=1 TO 30:POKE SS,90:S=USR(0)
1560 NEXT:PRINT@A+2,STRING$(52,32);
1565 PRINT@837,CHR$(30);:PRINT@837,;
1570 FOR TT=1 TO 200:NEXT:POKE SS,175
1575 S=USR(0):IF R=1 THEN 1590
1580 GOTO 1610
1590 PRINT"INTRUDER: FRIENDLY..."
1600 GOTO 1620
1610 PRINT"INTRUDER: ALIEN..."
1620 FOR TT=1 TO 500:NEXT
1630 IF XC<=1 THEN XC=XC+1:GOTO 1565
1640 GOTO 1670
1650 PRINT@837,"INTRUDER: DESTROYED..."
1655 FOR TT=1 TO 20:POKE SS,TT+10
1656 S=USR(0):NEXT
```

```
1660 DS=1:PRINT@L,"   ";:GOTO 1550
1670 REM VARIABLE COUNT
1680 PRINT@837,CHR$(30):PRINT@837,;
1690 IF DS=1 THEN 1810
1700 IF R=1 THEN 1750
1710 AI=AI+1
1720 PRINT"YOU NOW HAVE";AI;"ALIENS ";
1730 PRINT"IN THE DOME."
1740 GOTO 1790
1750 FI=FI+1
1760 PRINT"YOU HAVE WISELY LET";FI;"
1770 PRINT@901,;
1780 PRINT"OF THE FRIENDS PASS."
1790 REM CONT IF DAYS >=1
1800 GOTO 1900
1810 REM DESTROYED
1820 IF R=1 THEN 1870
1830 AD=AD+1
1840 PRINT"YOU HAVE DESTROYED";AD;"ALIENS."
1850 GOTO 1900
1870 FD=FD+1
1880 PRINT FD;"FRIENDS HAVE NOW BEEN
     DESTROYED."
1890 REM
1900 REM DX FOR DAYS DECREASE
1910 REM IF LESS THAN OR EQUAL TO 75
1920 DX=RND(100)
1930 IF DX<=75 THEN DA=DA-1:DF=DF+1
1940 FOR TT=1 TO 1200:NEXT
1950 PRINT@837,STRING$(128,32);
1960 IF DA<=0 OR FD>=10 THEN 1980
1970 FOR TT=1 TO 20.POKE SS,20:S=USR(0)
1975 NEXT:GOTO 860
1980 PRINT@837,;:IF FD>=10 THEN 2010
1990 PRINT"YOUR COMMISSION HAS EXPIRED..."
2000 GOTO 2020
2010 PRINT"YOUR COMMISSION IS TERMINATED..."
2020 FOR TT=1 TO 2000:NEXT
2030 CLS
2040 PRINT"DURING YOUR";DF;"DAY COMMISSION"
2050 PRINT"YOUR RECORD IS AS FOLLOWS:";P$;
2060 PRINT","
2070 PRINT"ALIENS ALLOWED TO PASS INTO DOME:";
2080 PRINT AI
2090 PRINT"ALIENS DESTROYED: ";AD
2100 PRINT"FRIENDLY INTRUDERS DESTROYED:";FD
2110 PRINT
2120 IF FD>=10 THEN 2240
```

```
2130 PRINT"YOUR COMMISSION HAS BEEN TERMED:"
2140 IF AD>=1 AND AD<=5 THEN 2170
2150 IF AD>=5 AND AD<=9 THEN 2190
2160 IF AD>=10 THEN 2210
2170 PRINT"IMPROPER. POOR USE OF LASER."
2180 GOTO 2270
2190 PRINT"FAIR. LASER USED, BUT NOT
     ENOUGH."
2200 GOTO 2270
2210 PRINT"EXCELLENT. ALL JUDGEMENTS WERE"
2220 PRINT"MADE WITH UTMOST CAUTION."
2230 GOTO 2270
2240 PRINT"YOU HAVE DESTROYED";FD;"FRIENDLY"
2250 PRINT"INTRUDERS ";P$;" DO NOT ATTEMPT"
2260 PRINT"TO RETURN TO THIS COMMISSION."
2270 PRINT
2280 PRINT"TERMINATION OF PROGRAM."
2290 END
```

Variables and Strings

SS - Sound location
P$ - User's last name
X$ - Crossroads graphics
A,J,AA - For graphics
A$,AA$ - Laser cannons, left and right
F$(1-5) - Figures, friendly
A$(1-5) - Figures, alien
K1,K2 - For blanking laser after firing
DA - Total days on commission
AI - Alien intruders in dome
FI - Friendly intruders in dome
AD - Aliens destroyed
FD - Friendly intruders destroyed
C$,CC$,L$,LL$ - For status message printout
L - Start location, intruder
I,N - Random amounts which determine when an intruder enters the dome
T - For-next loop, for above variables
X - For message printout (flashing)
LF - For noting if keys have been pressed in correct sequence
XC - For message printout (flashing)
DS - Intruders destroyed, alien or friendly
YU - Counter, figure up at highest point

R - Random selection: alien or friendly
E - Key closure by user
S - USR call for sound output
DX - Random variable. If less than or equal to 75 DA (days on commission) it will decrease by 1
DF - For message indicating the number of days on commission

Explanation of Program Lines

Lines 10-520 display the general instructions.

Line 530 has the user input his or her last name.

Lines 540-620 create the graphics for the crossroads and the four large squares.

Lines 630-690 establish the graphics for the lasers and display general messages.

Lines 700-780 establish the strings for the friendly and alien figures.

Lines 790-830 double the size of each figure. The variables set in line 830 are used for blanking out the laser after firing.

Lines 840-900 set more variables and print messages and values on the video.

Lines 910-920 determine the location the figure will start at.

Lines 930-990 maintain the present condition until N= I. The program will then branch to an alert and complete alert will be sounded when T= 200 in line 1005.

Line 1010 sets more variables before the figure is moved.

Line 1020-1070 present the alert message with the appropriate sounds.

Line 1080 randomly selects whether intruder is alien or friendly.

Lines 1100-1110 print the figure that was selected.

Line 1115 increases the counter YU by 1.

Line 1120 tests the value of E for key closure.

Line 1125 tests the value of LF. If LF is equal to 4, laser is ready to fire.

Lines 1130-1190 move the figure up or down, depending on the value of MM.

Lines 1200-1220 check the key closed with the value of LF. The only way the laser will fire is if all keys are pressed in the proper sequence.

Lines 1230-1270 increases the value of LF with each key closure. If E is equal to none of the values specified (line 1230), it

will equal 0 (zero), and the movement of the figure will continue at line 1090.

Lines 1360-1470 cause the firing of the laser with a sound output.

Line 1420 tests the value of the peek location, to see if figure was hit. If argument is true, the program will branch to line 1650.

Lines 1440-1500 display a message indicating that there was an indirect hit to the figure and produced a sound. The laser will then be blanked out while the figure continues moving. The laser is then ready to accept another key sequence to fire. All of this will take place in split seconds.

Lines 1520-1640 display a message indicating that an intruder has come into the dome.

Lines 1650-1660 display a message indicating that the alien has been destroyed.

Lines 1670-1890 display messages indicating how many aliens are in the dome, how many the user destroyed, and how many of those destroyed were friendly.

Lines 1900-1975 decrease the number of days in the user's present commission if DX (random) is less than or equal to 75. Line 1960 tests the values of both DA (days) and FD (friendly intruders destroyed). If either condition is met, the program is terminated.

Lines 1980-2290 are used when the user's commission has expired. All relevant figures are displayed, and the user's performance is evaluated. If the user has destroyed 10 friendly intruders, he or she cannot come out ahead.

Appendix A
Program Listings for the IBM PC

The following listings have been created especially for use on the IBM PC. Note that the caps lock key must be in effect in order to use the games. Also note that you should enter YES or NO, rather than Y or N unless the prompt indicates otherwise.

```
  1 REM program title:HOLD TIME
  3 CLEAR 350
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
    number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
100 WIDTH 40:COLOR 15,9,14
105 CLS:KEY OFF
125 PRINT TAB(15)"HOLD TIME":GOSUB 1050
130 PRINT:INPUT "INSTRUCTIONS";Z$
135 IF Z$="N" OR Z$="NO" THEN CLS:GOTO 275
140 PRINT:CLS
145 PRINT "HOLD TIME...A MATTER OF HIGH OR"
150 PRINT "LOW POINT TOTALS.  THE COMPUTER'S"
155 PRINT "CRAFT WILL BLIP ALL AROUND YOUR"
160 PRINT "VIDEO.  DOWN THE LEFT AND RIGHT"
165 PRINT "SIDES WILL BE ASTERISKS.  THESE"
170 PRINT "ARE YOUR LASERS.  AT THE INSTANT"
175 PRINT "YOU THINK THE CRAFT IS IN LINE"
180 PRINT "WITH THESE, HOLD THE SPACE BAR"
185 PRINT "YOUR LASERS WILL START MOVING"
190 PRINT "TOWARD THE CENTER.  THE COMPUTER"
195 PRINT "CRAFT WILL ALSO CONTINUE TO MOVE."
200 PRINT:GOSUB 1000:INPUT "PRESS ENTER TO CONTINUE"
    ,X$:CLS:PRINT
205 PRINT "YOU CAN 'HOLD' THE COMPUTER'S"
210 PRINT "CRAFT FROM MOVING FOR AN AMOUNT"
215 PRINT "OF TIME, ROUGHLY 20 SECONDS."
220 PRINT "POINTS WILL RANGE FROM 1500 TO "
225 PRINT "400, DEPENDING ON WHERE YOU"
230 PRINT "DESTROY THE CRAFT (TOP OR BOTTOM)"
235 PRINT "ON THE SCREEN.  TIME LIMIT FOR THE"
240 PRINT "ENTIRE RUN WILL BE JUST ABOUT"
245 PRINT "FIVE MINUTES.  THIS WILL BE NOTED"
250 PRINT "AT THE LOWER RIGHT OF YOUR SCREEN"
255 PRINT "COUNT STARTS AT 30."
260 PRINT "TO 'HOLD' THE COMPUTER'S"
265 PRINT "CRAFT, PRESS THE <CTRL> KEY":GOSUB 1025
270 PRINT
275 PRINT:INPUT "PRESS ENTER TO BEGIN",X$:CLS
295 DEF SEG=&H40:POKE &H17,0:C$="<X>"
300 REM outer graphics points
305 Q$=STRING$(38,42):MR=8:MC=16:MMR=1:MMC=1
310 G=0:PI=16:LOCATE 1,1:PRINT Q$:LOCATE 16,1:
    PRINT Q$
```

```
315 LOCATE 2,34:PRINT "20";:FOR XR=2 TO 15:XC=1
320 LOCATE XR,XC:PRINT LEFT$(Q$,1);:SOUND 10*
    (50-XR),1
325 PRINT TAB(40) LEFT$(Q$,1);:GOTO 500
330 IF MMR<>1 AND MMC<>1 THEN GOSUB 355
333 NEXT
335 LOCATE 1,1:PRINT STRING$(40,32);:LOCATE 16,1:
    PRINT STRING$(40,32)
340 FOR XR=2 TO 15:XC=1
345 LOCATE XR,XC:PRINT CHR$(32);:PRINT TAB(40)CHR$
    (32);:SOUND 70-XR,3
350 GOSUB 355:NEXT:GOSUB 355:GOTO 310
355 LOCATE 16,35:PRINT ABS(INT(TT/60)-30);
356 LOCATE MMR,MMC:PRINT STRING$(5,32);
360 IF MR<=2 OR MR>=15 THEN MR=7
361 COLOR 10:LOCATE MR,MC:PRINT C$;:TT=TT+1:COLOR 15
362 IF TT>=1800 THEN 830
365  MMR=MR:MMC=MC:DEF SEG=&H40:W=PEEK(&H17)
366 IF W=4 AND G<=20 THEN G=G+.4:GOSUB 770:RETURN
370 XX=INT(RND*4+1):REM craft move
375 ON XX GOTO 380,390,400,410
380 REM left move
385 MC=MC-INT(RND*5+1):GOTO 420
390 REM right move
395 MC=MC+INT(RND*5+1):GOTO 420
400 REM up move
405 MR=MR-1:GOTO 420
410 REM down move
415 MR=MR+1:DEF SEG=&HB800:IF PEEK(82)=32 THEN GOSUB
    700   'location of message from lines 745-762
416 IF PEEK(146)=32 THEN GOSUB 700    'location of
    message from line 315
417 IF PEEK(1126)=32 THEN GOSUB 700:    'location of
    message from line 705
418 IF PEEK(1186)=32 THEN GOSUB 700:    'location of
    message from line 715
420 IF MC<=3 THEN MC=40:MR=MR-1
422 IF MR<=2 AND MC<=2 THEN 410
425 IF MC>=35 THEN MC=1:MR=MR+1
427 IF MR>=15 AND MC>=35 THEN 400
430 SOUND 440,3:RETURN
500 REM fire
505 IF XR<=14 THEN W$=INKEY$
510 PI=PI-1:IF W$="" THEN 330
515 IF W$=" " THEN 525
520 GOTO 330
525 REM laser advance
```

```
530 IF XR>=13 AND XC>=30 THEN 330
532 IF XR<=2 AND XC<=40 THEN 330
535 DEF SEG=&HB800:HR=XR:HC=XC:H1=(HR-1)*80:
    H2=H1+78:UN=UN+1:S2=1000
536 IF RP=1 THEN LOCATE 2,20:PRINT ABS(R-UN);
540 DEF SEG=&HB800:POKE H1,32:POKE H2,32:
545  IF PEEK(H1+2)<>32 OR PEEK(H2-2)<>32 THEN 570
550 H1=H1+2:H2=H2-2:S2=S2-10:SOUND S2,3
555 POKE H1,42:POKE H2,42
560 IF ABS(H2-H1)<=4 THEN 900
565 GOSUB 355:GOTO 540
570 REM direct hit
575 POKE H1,42
580 IF PEEK(H2-2)<>32 THEN 600
585 POKE H2,32:H2=H2-8
590 POKE H2,42
595 FOR O=1 TO 2:SOUND 1000,3:SOUND 800,3:NEXT:
    GOTO 580
600 REM delete
605 Z=1:POKE H1,32:POKE H2,32
610 N=(MR-1)*80+(MC-1)*2:S1=2200
615 DEF SEG=&HB800:POKE N,32:POKE N+2,32
620 POKE N+4,32
625 J=2*INT(RND*5)
630 IF N+J-80>=158 THEN 640
635 GOTO 660
640 POKE N+J-80,42:POKE N+J+80-J,42
645 FOR I=1 TO J*10:NEXT
650  POKE N+J-80,32:POKE N+J+80-J,32
655 GOTO 675
660 POKE N+J+80,42:POKE N+J+80-J,42
665 FOR I=1 TO J*10:NEXT
670 POKE N+J+80,32:POKE N+J+80-J,32
675 IF Z<=7 THEN 685
676 SOUND 1000,3
680  IF RP=0 THEN GOTO 690 ELSE PI=PI+1:GOTO 695
685 Z=Z+1:SOUND 220,3:GOTO 625
690  PI=PI+1:X$=X$+"+"
695 PP=PP+PI*100:GOSUB 700:GOTO 900
700 A$="TOTAL POINTS:":T$="TIME LAPSE:"
705 FR=15:FC=3
710 FOR I=1 TO LEN(A$)
715 LOCATE FR,FC+25:PRINT(MID$(T$,I,1));
716 LOCATE FR,FC:PRINT MID$(A$,I,1);
720 FC=FC+1:NEXT
725  PRINT PP;:IF RP=1 THEN 735
730 IF X$="" THEN RETURN
```

```
735  REM reserve power
740 R$="RESERVE POWER:"
745  FR=2:FC=2
750 FOR I=1 TO LEN(R$)
755  LOCATE FR,FC:PRINT MID$(R$,I,1);
760 FC=FC+1:NEXT
762 IF RP=1 THEN X$=STR$(ABS(R-UN))
765  PRINT X$;
770 TI$="HOLD TIME:":REM hold time
775 DEF SEG=&HB800:IF PEEK(1186)<>32 THEN 790
780 FR=2:FC=25:FOR I=1 TO LEN(TI$)
785 LOCATE FR,FC:PRINT MID$(TI$,I,1);:FC=FC+1:NEXT
790 LOCATE 2,36:PRINT INT(ABS(G-20));
795 RETURN
800 REM time remaining
805 IF RP=1 THEN DK=1:GOTO 910
810 IF TT>=1800 THEN 830
820 GOTO 300
830 FOR TI=1 TO 1200:NEXT
835 CLS:R=LEN(X$)
840 IF R=0 THEN 960
845 PRINT "VERY WELL DONE...."
850 PRINT "YOU HAVE ";R;" UNITS OF"
855 PRINT "RESERVE POWER...."
860 PRINT "PRESS ENTER TO USE";
865 INPUT X$:TT=0:RP=1:UN=0
870 CLS
875 GOTO 300
900 DEF SEG=&H40:POKE &H17,0:REM delete reserve
    units
905 IF RP=0 THEN 330
910 IF UN>=R THEN 925
915 IF DK=0 THEN 330
920 DK=0:GOTO 810
925 REM finish
930 FOR TI=1 TO 1200:NEXT
935 CLS
940 PRINT "RESERVE POWER EXHAUSTED....."
945 PRINT "TOTAL POINTS ACCUMULATED: ";PP
950 PRINT
955 GOTO 980
960 PRINT "YOU HAVE DESTROYED NOTHING"
965 PRINT "YOU HAVE NO RESERVE POWER."
970 PRINT
975 GOTO 945
980 PRINT
985 PRINT "END OF RUN."
990 GOSUB 1025:END
```

```
1000 OUTERSPACE$="MN O1 L1 F O2 C ML F L2 F P8 L8 A"
1010 PLAY "MB XOUTERSPACE$;L1 G+ L2 G+ P8 L8 G+ L1 A
     L2 A;"
1020 RETURN
1025 OUTERSPACE$="MN O1 L1 F O2 C ML F L2 F P8 L8 A"
1030 PLAY "MB XOUTERSPACE$;L1 B- L8 B- O3 D E L4 F G
     L1 A L2 A;"
1040 RETURN
1050 BACKGROUND$="MB MS O1 L4 F C F C F C F C"
1060 PLAY "MB XBACKGROUND$;"
1070 RETURN
```

```
  1 REM program title: elevator
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
    number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 20 CLS:KEY OFF:WIDTH 40:W$="$$#####.##"
 25 DEF SEG=&HB800
 30 DIM F(60),F$(60),G$(60),GK(60)
 50 PRINT TAB(10)"*** ELEVATOR ***"
 60 PRINT
 70 BEEP:PRINT "INSTRUCTIONS REQUIRED (Y/N)";
 80 INPUT I$:IF I$="N" THEN N=1
 90 BEEP:INPUT "FIRST NAME";N$
100 PRINT:IF N=1 THEN 230
110 BEEP:INPUT "PRESENT OCCUPATION";P$
120 PRINT "YOU HAVE JUST BEEN FIRED AS A ";
125 PRINT P$;"!!"
130 PRINT "BUT, YOU ARE IN LUCK!! I HAVE JUST"
140 PRINT "HIRED YOU TO BE AN ELEVATOR ATTENDANT"
150 PRINT "IN MY COMPUTERIZED DEPARTMENT STORE."
160 PRINT "YOUR JOB WILL BE SIMPLE, ";N$;". ALL"
170 PRINT "YOU HAVE TO DO IS REMEMBER WHAT IS"
180 PRINT "CONTAINED ON EACH OF SIXTEEN FLOORS"
190 PRINT "SO THAT YOU CAN TAKE ALL OF MY CUSTOMERS"
200 PRINT "TO THE RIGHT LOCATION..."
210 PRINT
220 PRINT
230 REM what's on each floor?
240 FOR I=2 TO 19:READ F$(I):NEXT
250 IF N=1 THEN 370 ELSE GOSUB 2150:PRINT
260 PRINT "I ALMOST FORGOT TO TELL YOU--THE"
270 PRINT "CUSTOMERS, WHEN STEPPING ONTO THE "
280 PRINT "ELEVATOR, WILL TELL YOU WHAT THEY ARE"
290 PRINT "SHOPPING FOR.  THEY WON'T TELL YOU "
300 PRINT "AGAIN, BUT FOR EVERY CUSTOMER YOU PLACE"
310 PRINT "ON THE RIGHT FLOOR, YOU'LL BE PAID "
320 PRINT "$10.00.  THAT WILL BE YOUR ONLY WAGE!"
330 PRINT "CUSTOMERS WILL BE GETTING ON AND OFF THE"
340 PRINT "ELEVATOR AT DIFFERENT STOPS, AND YOU "
350 PRINT "WILL NEED TO REMEMBER THESE, TOO, "
355 PRINT N$;"!"
360 GOSUB 2150
370 PRINT "HERE ARE THE FLOORS AND CONTENTS:"
380 FOR I=2 TO 16
390 SOUND 50*I+100,3:PRINT I;") ";F$(I);
```

```
395 IF I=8 THEN TT=I*400:PRINT:GOSUB 2190
400 IF I>=2 AND I<=5 THEN 440
410 IF I>=6 AND I<=10 THEN PRINT :GOTO 470
420 IF I>=11 AND I<=14 THEN 450
430 IF I>=15 AND I<=16 THEN 460
440 PRINT " ";F$(17):GOTO 470
450 PRINT " ";F$(18):GOTO 470
460 PRINT " ";F$(19)
470 IF RX=1 THEN RETURN ELSE NEXT
475 REM variable rx for print elements
480 FOR I=1 TO 6:PRINT:NEXT:TT=TT+2800
490 GOSUB 2190:CLS:C=0:GK=0
500 SOUND 440,3:PRINT "YOU SHOULD BE ALL SET, ";N$
510 TT=TT-1800:GOSUB 2190:RX=1
520 REM get customers
530 C=INT(RND*10+1)
540 PRINT N$;" YOU NOW HAVE";C;"CUSTOMER(S)"
550 PRINT "GETTING ONTO MY ELEVATOR.":FOR KO=1 TO
    C:SOUND KO*100,2:NEXT
560 REM get c responses
570 I=1
580 REM floor number
590 F=INT(RND*14+2)
600 F(I)=F:I=I+1:GK(F)=GK(F)+1
610 IF I<=C THEN 590
620 REM floor titles
630 FOR I=1 TO C
640 G$(I)=F$(F(I))
650 NEXT
660 GOSUB 2190:PRINT
670 PRINT "THE CUSTOMERS' DESTINATIONS:":FL=1
680 PRINT:FOR I=FL TO C
690 PRINT G$(I);"";
700 F=F(I)
710 IF F>=2 AND F<=5 THEN 760
720 IF F>=11 AND F<=14 THEN 770
730 IF F>=15 AND F<=16 THEN 780
740 PRINT " FLOOR"
750 NEXT:GOTO 790
760 GOSUB 440:GOTO 750
770 GOSUB 450:GOTO 750
780 GOSUB 460:GOTO 750
790 GOSUB 2150
800 CLS
810 SOUND 100,1:PRINT "PRESS <U> FOR <UP>"
820 SOUND 200,1:PRINT "PRESS <D> FOR <DOWN>"
830 PRINT "THEN ENTER THE FLOOR NUMBER"
```

```
 840 NN=0:REM direction
 850 GOSUB 2160
 860 D$=A$:IF D$="U" THEN D$="UP":GOTO 870
 865 IF D$="D" THEN D$="DOWN"
 868 GOTO 870
 870 SOUND 100,1:PRINT "DIRECTION: ";D$
 880 SOUND 200,1:PRINT "FLOOR NUMBER";
 890 INPUT FR:SOUND 100+25*FR,3
 895 IF FR=1 AND TF=0 THEN 1280
 900 IF FR<1 OR FR>16 THEN 800
 905 IF FR=TF THEN 1280
 910 IF FR<TF AND D$="UP" THEN 1260
 915 IF FR>TF AND D$="DOWN" THEN V=1:GOTO 1260
 920 GOSUB 2190:CLS:REM pause before start
 930 IF D$="UP" THEN 970
 940 IF D$="DOWN" THEN 1040
 950 BEEP:PRINT "SELECT A DIRECTION, ";N$;"!!"
 960 PRINT:GOTO 810
 970 REM up
 980 IF TF>=17 THEN 1030
 990 GOTO 1300
1000 SOUND 200,3:SOUND 100,4:PRINT "YOU ARE ON THE
     ";T$;" FLOOR, ";N$
1010 PRINT "YOU CAN NOT GO ANY ";H$;"!!"
1020 GOTO 960
1030 T$="TOP":H$="HIGHER":GOTO 1000
1040 IF TF<=1 THEN 1060
1050 GOTO 1070
1060 T$="BOTTOM":H$="LOWER":GOTO 1000
1070 REM down continued movement
1080 FS=P+640:NN=1
1085 IF FF>=58 AND FF<=64 THEN 1220
1090 POKE P,FF:SOUND 100*FF-4500,1
1100 FOR T=1 TO 40:NEXT
1110 POKE P,32:POKE P+2,32:P=P+80
1120 IF P<>FS THEN 1085
1130 REM continue if not on floor
1140 IF TF<>FR THEN 1160
1150 GOTO 1500
1160 FS=FS+560
1170 IF P>=1244 THEN 1190
1180 GOTO 1085
1190 POKE P,32:POKE P+2,32:P=280
1200 FS=P+640:FF=FF-1:TF=TF-1
1210 GOTO 1085
1220 REM floors above 9
1230 POKE P,49:POKE P+2,FF-10:SOUND 100*FF-4500,1
```

```
1240 IF NN=1 THEN 1100
1250 GOTO 1340
1260 PRINT "IF WANTING TO GO ";D$;", USE A"
1265 IF V=1 THEN V=0:PRINT "USE A LOWER NUMBER."
     :GOTO 1275
1270 PRINT "HIGHER NUMBER THAN THIS FLOOR."
1275 GOTO 1290
1280 PRINT "YOU ARE ON THAT FLOOR, ";N$;"!!"
1290 GOSUB 2190:GOTO 800
1300 REM up movement
1310 IF TF=0 THEN 1460
1320 FS=P-640:IF FS<=278 THEN FS=280
1325 IF FF>=58 AND FF<=64 THEN 1220
1330 POKE P,FF:SOUND 100*FF-4500,1
1340 FOR T=1 TO 50:NEXT
1350 POKE P,32:POKE P+2,32:P=P-80
1360 IF P<>FS THEN 1325
1370 REM continue if not on floor
1380 IF TF<>FR THEN 1400
1390 GOTO 1500
1400 FS=FS-560
1410 IF P<=278 THEN 1430
1420 GOTO 1325
1430 POKE P,32:POKE P+2,32:P=1240
1440 FS=P-640:FF=FF+1:TF=TF+1
1450 GOTO 1325
1460 TF=1:P=920:FF=49
1470 FS=P-560:GOTO 1325
1500 REM on floor requested
1510 IF FF>=58 AND FF<=64 THEN 1530
1520 POKE P,FF:GOTO 1540
1530 POKE P,49:POKE P+2,FF-10
1540 FOR I=1 TO C
1550 IF FR=F(I) THEN 1570
1560 NEXT:IF GK(FR)>=1 THEN XX=1:PA=PA+10*GK(FR):
     KG=KG+GK(FR):GK(FR)=0
1565 GOTO 1600
1570 PA=PA+10:KG=KG+1:GK(FR)=GK(FR)-1
1580 F(I)=0:XX=1
1590 GOTO 1560
1600 REM did customers want off here
1610 IF XX=0 THEN 1630
1620 XX=0:GOTO 1770
1630 PRINT N$;", NO CUSTOMERS WANTED TO GET OFF":
     SOUND 200,3:SOUND 100,4
1640 PRINT "ON THIS FLOOR...":NC=NC+1
1645 HJ=HJ+1:IF HJ>=4 THEN HJ=0:FZ=1
1650 REM more customers stepping on
```

```
1655 AA=INT(RND*2+1):IF JV=1 THEN AA=1:FZ=0
1660 IF FZ=1 OR FR=1 THEN AA=1:FZ=0
1670 IF AA=1 THEN 1690
1680 GOTO 1730
1690 YU=INT(RND*5+1):YY=C:C=C+YU:C2=YU
1700 IF C>=59 THEN JV=1:C=C-C2:GOTO 1730
1710 REM get destinations
1720 GOTO 1800
1730 REM continue if not finished
1740 KG=0:IF GK>=C THEN 1940
1750 GOSUB 2190
1760 GOTO 800
1770 FOR KO=1 TO KG:SOUND KO*100,2:NEXT:PRINT KG;
     "OF THE CUSTOMERS JUST"
1780 PRINT "STEPPED OFF MY ELEVATOR, ";
1785 PRINT N$;".":GOSUB 2700
1790 GK=GK+KG:GOTO 1650
1800 REM yy more destinations
1810 YU=ABS(C-YY):IF GK>C OR GK>=10 THEN 1940
1820 PRINT N$;", YOU JUST HAD";YU;"MORE"
1830 PRINT "CUSTOMERS STEP ONTO THE"
1840 PRINT "ELEVATOR.  THE DESTINATION(S):"
1850 KG=0:PRINT:FOR I=YY+1 TO C
1860 F=INT(RND*(16)+1)
1865 IF F=TF THEN 1860
1870 IF F=1 THEN F$(1)="GROUND":FZ=1
1880 F(I)=F:NEXT:REM floors
1890 REM title of floors
1900 FOR I=YY+1 TO C
1910 G$(I)=F$(F(I)):NEXT
1920 FL=YY+1
1930 GOTO 680
1940 REM stop/cont
1950  PRINT:PRINT "YOU HAVE NOW COLLECTED:"
1960 PRINT USING W$;PA:PRINT:GOSUB 2190
1965 PRINT:GOSUB 2500
1970 IF JV=1 OR GK>=C THEN 1990
1980 GOSUB 2190:GOTO 800:REM all not off
1990 PRINT:GOTO 2300
2000 REM data for floors
2010 DATA WOMENS,MENS,CHILDRENS,INFANTS
2020 DATA TOYS,CARPETS,COMPUTERS
2030 DATA APPLIANCES,FURNITURE,HARDWARE
2040 DATA HOBBY,AUTO,OFFICE
2050 DATA BATHROOM,BEDROOM
2060 DATA CLOTHES,SUPPLIES,ITEMS
2070 REM data element 17 will be used
2080 REM with elements 2-5
```

```
2090 REM data element 18 will be used
2100 REM with elements 11-14
2110 REM data element 19 will be used
2120 REM with elements 15 & 16
2150 PRINT:PRINT "PRESS A KEY...":SOUND 440,3
2160 A$=INKEY$
2170 IF A$="" THEN 2160
2180 RETURN
2190 REM time loop
2200 FOR T=1 TO TT:NEXT
2210 GOTO 2180
2300 REM continue?
2310 PRINT "WOULD YOU LIKE TO TAKE ON"
2320 PRINT "SOME MORE CUSTOMERS, ";N$;"?"
2330 PRINT "YOUR TOTAL EARNINGS:";
2340 PRINT USING W$;PA;
2350 INPUT A$
2360 IF A$<>"Y" AND A$<>"YES" THEN 2400
2370 TT=2000:JV=0:PRINT
2380 PRINT "YOU ARE ON FLOOR ";TF
2390 NC=0:GOTO 490
2400 PRINT
2410 PRINT "END OF ELEVATOR"
2420 END
2500 REM point scale/pay deduction
2510 IF NC<=0 THEN NC=0:RETURN
2520 PRINT "YOU MADE";NC;"STOPS THAT WERE"
2530 PRINT "NOT NECESSARY, ";N$;".  YOU"
2540 PRINT "WILL TAKE A DEDUCTION IN PAY"
2550 PRINT "OF $20.00 FOR EACH ONE OF"
2560 PRINT "THOSE STOPS, TOTAL";
2570 NN=NC*20:PRINT USING W$;NN
2580 PRINT "LEAVING YOU";
2590 PA=PA-NN
2600 PRINT USING W$;PA:GOSUB 2190
2610 RETURN
2700 REM any tips?
2710 IF KG<=1 THEN RETURN
2720 AA=INT(RND*2+1)
2730 IF AA=1 THEN RETURN
2740 TI=INT(RND*100+1)
2750 PA=PA+TI
2760 PRINT
2770 PRINT "YOU'RE IN LUCK ";N$;", A"
2780 PRINT "NICE CUSTOMER JUST GAVE "
2790 PRINT "YOU A TIP OF";
2800 PRINT USING W$;TI
2810 GOSUB 2190:RETURN
```

```
  1 REM program title:FAREWELL
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
    number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 20 WIDTH 40:KEY OFF:COLOR 1,14,11:CLS:PRINT TAB(12)
    "*** FAREWELL ***":GOSUB 1350
 25 PRINT
 30 INPUT "NUMBER PLAYING (1-10)";A
 35 IF A<1 OR A>10 THEN 25
 40 PRINT "THE ";A;" PLAYERS'FIRST NAMES":PLAY "O1A"
 45 FOR I=1 TO A
 50 INPUT Q$(I):SOUND 50+20*I,3:NEXT
 55 PRINT:U=1
 60 PRINT "ANY OF YOU NEED TO SEE"
 65 INPUT "THE INSTRUCTIONS";X$
 70 IF X$<>"Y" AND X$<>"YES" THEN 210
 75 PRINT
 80 PRINT "READ CAREFULLY, IT COULD MEAN"
 85 PRINT "YOUR LIFE (OR THE GAME!!)..."
 90 PRINT "EACH PLAYER WILL SEE AND STUDY A"
 95 PRINT "MATRIX OF 6 BY 5 SQUARES (POSITIONS)"
100 PRINT "THESE WILL RANGE FROM THE "
105 PRINT "LETTERS A THROUGH Z WITH"
110 PRINT "FOUR SPACES BLANK.  TO PLAY:"
115 PRINT "THE COMPUTER WILL SELECT FIVE"
120 PRINT "POSITIONS AND PLACE COBRAS"
125 PRINT "WITHIN EACH.  THE COMPUTER WILL"
130 PRINT "SHOW YOU WHERE THREE OF THE COBRAS ARE"
135 GOSUB 1340:PRINT:INPUT "PRESS ENTER,PLEASE:";X$
140  CLS
145 PRINT "NOW THE COMPUTER WILL SELECT A"
150 PRINT "POSITION AND FLASH AN ASTERISK"
155 PRINT "THERE.  EACH PLAYER MUST ENTER FROM"
160 PRINT "MEMORY THE NUMBER WHICH WAS IN THAT"
165 PRINT "POSITION.  IF CORRECT, THE PLAYER"
170 PRINT "GAINS 1000 POINTS; IF INCORRECT, THE"
175 PRINT "PLAYER LOSES 1000 POINTS. AND...IF YOU"
180 PRINT "SELECT A POSITION WHERE THERE'S A "
185 PRINT "COBRA...YOU'RE OUT OF THE GAME, DEAD!!"
190 PRINT "SELECTION WILL CONTINUE FOR AS LONG AS"
195 PRINT "EACH PLAYER WANTS OR UNTIL TWENTY-ONE"
205 PRINT "POSITIONS HAVE BEEN ELIMINATED."
210 GOSUB 1350:GOSUB 220:INPUT "PRESS ENTER TO
    START";X$
```

```
215 CLS:GOTO 300
220 DIM L$(31),PR(31),PC(31),L(31)
225 REM letters?
228 IF SL<>0 THEN PRINT "ARRANGING LETTERS..."
230 L=65:I=1:PZ=1000
235 L$(I)=CHR$(L):L(I)=I
240 IF I=27 THEN 255
245 I=I+1:L=L+1:GOTO 235
250 GOTO 235
255 REM sort letters randomly
260 J=1
265 N=INT(RND*(I-1)+1)
270 IF L(N)=0 THEN 265
275 L$(J)=CHR$(N+64)
280 L(N)=0:J=J+1
285 IF J=I THEN RETURN
290 GOTO 265
300 REM positions
305 PC=5:PR=5:I=1:PL=PR+2:J=1:PK=PC
310 PR(I)=PR:PC(I)=PC:L(I)=I
315 PC=PC+6:I=I+1:J=J+1
320 IF I=27 THEN 345
325 IF J=7 THEN 335
330 GOTO 310
335 PR=PL:PL=PL+2:J=1:PC=PK
340 GOTO 310
345 REM the remaining four
350 L$(I)=" ":PR(I)=PR:PC(I)=PC:L(I)=I
355 PC=PC+6:I=I+1:J=J+1
360 IF J=7 THEN 370
365 GOTO 350
370 REM move the four blanks
375 I=1:J=1:L=27
380 N=INT(RND*26+1)
385 IF L(N)=0 THEN 380
390 F$=L$(N):L$(N)=L$(L)
395 L$(L)=F$:L(N)=0
400 L=L+1:IF L=31 THEN 410
405 GOTO 380
410 REM now print letters
415 CLS:FOR I=1 TO 30
420 LOCATE PR(I),PC(I):PRINT L$(I):SOUND 10*
    (ASC(L$(I))),1:NEXT
425 REM final graphics
430 X=2
435 FOR Y=5 TO 13
440 LOCATE Y,X:PRINT CHR$(222):NEXT
```

```
445 X=X+6
450 IF X>=40 THEN 460
455 GOTO 435
460 FOR X=2 TO 38
465 LOCATE Y,X:PRINT CHR$(223):NEXT
470 Y=Y-2
480 IF Y<=3 THEN 490
485 GOTO 460
490 REM cobra positions/player
495 C=1:LOCATE 15,5:PRINT "PLAYER: ";Q$(U);
500 N=INT(RND*(I-5)+1)
505 IF L(N)=0 OR L(N)=100 THEN 500
510 CBR(C)=PR(N):CBC(C)=PC(N):F$=L$(N):C$(C)=F$
515 L$(N)=CHR$(42):L(N)=100
520 IF C=5 THEN 560 ELSE IF C<=3 THEN GOSUB 530
525 C=C+1:GOTO 500
530 XX=0          'define 3 (1 for computer)
535 LOCATE PR(N),PC(N):PRINT L$(N);:FOR T=1 TO
    100:NEXT
540 LOCATE PR(N),PC(N):PRINT " ";:FOR T=1 TO 50:NEXT
545 LOCATE PR(N),PC(N):PRINT F$;
550 IF XX<=0 THEN XX=XX+1:GOTO 535
555 RETURN
560 REM computer select/if not done
565 SL=1
575 IF SL>=22 THEN 1000 ELSE GOSUB 1200
585 LOCATE 16,5:PRINT "I WILL SELECT A POSITION"
    :SOUND 220,1
590 N=INT(RND*(L-1)+1):GOSUB 1240
595 IF L(N)=0 OR L(N)=100 THEN 590
600 F$=L$(N):IF SL=1 THEN GOSUB 530
605 C$=F$:NU=N:GOSUB 1300:GOSUB 1200
610 LOCATE 16,5:PRINT "SELECT A POSITION BY LETTER
    ONLY,";:SOUND 220,1
615 LOCATE 17,5:PRINT Q$(U);
620 INPUT L$
625 I=1
635 IF L$=L$(I) THEN 665
645 I=I+1
655 IF I>=31 THEN 675
660 GOTO 635
665 N=I:F$=L$:GOSUB 530
670 GOTO 800
675 REM cobra position or blank
680 GOSUB 1200
685 GOSUB 1200
690 FOR I=1 TO 5
```

```
695 IF L$=C$(I) THEN 735
705 NEXT
715 LOCATE 16,5:PRINT "YOU CANNOT SELECT A BLANK!!"
    :SOUND 220,1
725 GOSUB 1240:GOSUB 1200:GOTO 610
735 GOSUB 1340:REM cobra
745 LOCATE 16,5:PRINT "YOU'RE DEAD, ";Q$(U);" !!!"
755 LOCATE 17,5:PRINT "THAT IS A COBRA POSITION !!!"
765 LOCATE 18,5:PRINT "** FAREWELL **":GOSUB 1240
770 ZZ=1:GOSUB 1200:GOTO 1020
800 GOSUB 1240:GOSUB 1200:IF L$<>C$ THEN 900
805 GOSUB 1350:LOCATE 16,5:PRINT "CONGRATULATIONS "
    ;Q$(U);"..."
810 LOCATE 17,5:PRINT "YOU'VE JUST GAINED ";PZ;
815 PRINT "POINTS!!"
820 REM points/continue
825 W(U)=W(U)+PZ
830 LOCATE PR(NU),PC(NU):PRINT " ";
835 LOCATE 19,25:PRINT "TOTAL POINTS: ";W(U);
840 GOSUB 1240:GOSUB 1200
845 LOCATE 16,5:PRINT "YOU NOW HAVE ";W(U);
    "POINTS..."
850 LOCATE 17,5:PRINT "...CONTINUE (Y/N)";
855  INPUT X$:IF X$="N" THEN 1020
860 SL=SL+1:L(NU)=0:L$(NU)=" "
865 GOTO 575
900 REM wrong position
905 LOCATE 16,5:PRINT "WRONG POSITION,";Q$(U):SOUND
    220,1:SOUND 100,1
910 LOCATE 17,5:PRINT "YOU ARE NOW MINUS";PZ;
915 PRINT "POINTS...":GOSUB 1240
920 GOSUB 1200:W(U)=W(U)-PZ
925 LOCATE 19,25:PRINT "TOTAL POINTS: ";W(U);
930 GOTO 610
1000 REM final/another player
1005 GOSUB 1240:GOSUB 1200
1010 GOSUB 1350:LOCATE 16,5:PRINT "THAT'S IT, "
     ;Q$(U);". YOU'VE"
1015 LOCATE 17,5:PRINT "REACHED THE LIMIT..."
1020 LOCATE 18,5:PRINT "PRESS ENTER, PLEASE:";
1025 INPUT X$:CLS
1030 IF ZZ=1 THEN ZZ=0:GOTO 1040
1035 T(U)=W(U)
1040 U=U+1:IF U>A THEN 1100
1045 PRINT Q$(U);", YOU ARE NEXT..."
1050 PRINT "PRESS ENTER WHEN READY:";
1055 INPUT X$:CLS
```

```
1060 GOSUB 225:GOTO 300
1100 REM final/score results
1105 PRINT "PLAYERS AND POINTS:"
1110 FOR I=1 TO A
1115 PRINT Q$(I);": TOTAL POINTS,";T(I)
1120 NEXT
1125 PRINT
1130 PRINT "PROGRAM TERMINATED..."
1135 END
1200 REM clear print area
1210 LOCATE 16,5:PRINT STRING$(34,32)
1220 LOCATE 17,5:PRINT STRING$(34,32)
1230 RETURN
1240 REM time delay
1250 FOR TI=1 TO 2000:NEXT
1260 RETURN
1300 REM flash selection course
1305 FOR I=1 TO NU
1310 IF L$(I)="*" OR L$(I)=" " THEN 1330
1315 LOCATE PR(I),PC(I):PRINT "*";:SOUND 1000,1
1320 FOR TI=1 TO 20:NEXT TI
1325 LOCATE PR(I),PC(I):PRINT L$(I);:SOUND 1500,1
1330 NEXT I
1335 RETURN
1340 PLAY "MB O3L8CDL4E-DL8CP8CDE-GDE-C
1345 RETURN
1350 PLAY "MB MLL8O1AB-O2D-DD-O1B-O2L2D-"
1355 RETURN
```

```
 10 REM Program Title:  Mind Invasion--IBM PC
 20 REM Instruction Routine
 25 WIDTH 40
 30 CLEAR 500:COLOR 2,0,14:FL=1:GOSUB 1500:J=1:TR=1
 40 REM outer graphics
 50 DEF SEG=&HB800:X=1280:Y=1358:K=219:PN=1:TX=Y-600:
    TL=TX
 60 POKE X,K:POKE Y,K:POKE X+10,K:POKE Y-10,K
 70 X=(X-80+2):Y=(Y-80)-2:J=J+1
 80 IF J<>8 THEN 60
 90 REM goto character print if done
100 IF J=16 THEN N=42:POKE TX+1,4:POKE TX,2:GOTO 140
110 POKE X,K:POKE Y,K:POKE X+10,K:POKE Y-10,K
115 IF PN=1 THEN A(1)=X:A(2)=Y:L=A(1):R=A(2)
120 X=(X-80)-2:Y=(Y-80)+2
130 J=J+1:PN=2:GOTO 100
140 LOCATE 23,17:PRINT "YOUR POINTS=";
150 PI=0
160 GOSUB 910
170 LOCATE 22,1:PRINT "PLAYER --";N$(PZ)
200 REM character graphics variables
210 J=7:E=8:ET=1:F=8:FT=33:T=1:E2=E:ET2=ET:F2=F:FT2
    =FT:TC=1:TCT=1:BC=15:BCT=20
220 KY=1:G=20:GOSUB 380:IF TI>=3 THEN 880
230 LOCATE E,ET:PRINT W$;:LOCATE F,FT:PRINT W$;:E1
    =E+1:ET1=ET:F1=F+1:FT1=33
240 T=T+1:GOSUB 350
250 LOCATE E,ET:PRINT L$;:LOCATE F,FT:PRINT L$;
260 IF T>2 THEN J=J-1:GOSUB 380:GOSUB 350
270 COLOR 14:LOCATE (E-1),ET:PRINT W$:LOCATE E1,ET1:
    PRINT W$:COLOR 2
280 LOCATE(F-1),(FT-1):PRINT W$:LOCATE F1,FT1:
    PRINT W$
290 LOCATE (E-1),ET:PRINT L$;:LOCATE E1,ET1:PRINT L$;
300 LOCATE (F-1),(FT-1):PRINT L$;:LOCATE F1,FT1:PRINT
    L$;:GOTO 600
310 E=E-1:F=F-1:FT=FT-1:E1=E2+2:F1=F2+2:FT1=FT2+2
320 E2=E2+1:F2=F2+1:FT2=FT2+1:T=T+1
330 REM reset variables if done
335 GOTO 700
340 IF T>8 THEN 200 ELSE KY=KY+1:GOTO 260
350 REM for characters clear
360 L$=STRING$(J,255):Q$=STRING$(G,42)
370 Q1$=STRING$(G,32):RETURN
380 W$=STRING$(J,N)
```

```
390 RETURN
400 X$=INKEY$    'user movement
405 TI=TI+1      'limit count without movement
410 IF LEN(X$)=0 THEN 340
420 TI=1      'placement/limit/reset
425 IF LEN(X$)=2 THEN Y$=MID$(X$,2,1)
430 IF ASC(Y$)=72 THEN 460
440 IF ASC(Y$)=80 THEN 490
450 GOTO 340
460 POKE TX,32:TX=TX-80
470 IF ABS(TX-TL)>160 THEN TX=TX+80
480 POKE (TX+1),4:POKE TX,2:SOUND 880,1:U(KY)=ASC
    (Y$):GOTO 340
490 POKE TX,32:POKE TX+1,2:TX=TX+80
500 IF ABS(TX-TL)>160 THEN TX=TX-80
510 POKE (TX+1),4:POKE TX,2:SOUND 880,1:U(KY)=ASC
    (Y$):GOTO 340
600 REM top/bottom characters
610 COLOR 14:LOCATE TC,TCT:PRINT Q$;
620 LOCATE BC,BCT:PRINT Q$;:COLOR 2:FOR X=2000 TO
    1900 STEP -50:SOUND X,1:NEXT
630 LOCATE TC,TCT:PRINT Q1$;
640 LOCATE BC,BCT:PRINT Q1$;
650 TC=TC+1:TCT=TCT+1:BC=BC-1:BCT=BCT-1
660 REM replace user character
670 IF PEEK(TX)=32 THEN POKE TX+1,4:POKE TX,2:SOUND
    300,1
680 GOTO 310
700 REM crusher movement
705 POKE A(1)+10,32:POKE A(2)-10,32
710  POKE L,32:POKE R,32
720  IF ABS(L-R)=2 THEN 750
730 L=L+8:R=R-8
740 POKE L,K:POKE (L+10),K:GOTO 400
750 L=A(1):R=A(2):GOTO 800
760 POKE L,K:POKE (L+10),K:PLAY "MB XMIND$;"
770 POKE R,K:POKE (R-10),K
780 REM recycle
790 GOTO 400
800 REM crushed?
810 IF PEEK(TX)=32 THEN 840
820 IF TI<=2 AND U(KY)<>U(KY-1) THEN PI=PI+ABS
    (TX-TL):GOSUB 910
830 GOTO 760
840 REM yes...crushed
850 LOCATE 17,1:PRINT "<<CRUSHED>>";
855 PLAY "MB O4L32BGBGBGBGBGBGBGBG"
```

```
 860 POKE TX,32
 870 FOR LX=1 TO 2000:NEXT:GOTO 950
 880 REM limit reached/no movement
 890 TI=0
 900 PI=PI-ABS(TL-TX):GOSUB 910:GOTO 920
 910 LOCATE 23,31:PRINT PI;:RETURN
 920 GOTO 230
 950 REM player advance/end
 960 TP(PZ)=PI:PI=0:PZ=PZ+1
 970 IF PZ>PL THEN GOSUB 1200:TR=TR+1:PZ=1
 980 IF TR>=3 THEN GOSUB 1240:GOTO 1010
 990 FOR LX=1 TO 14:PRINT:NEXT
1000 GOSUB 2100:CLS:J=1:GOTO 40
1010 REM end/high point winner
1020 T=0:H=1:IF H=PL THEN 1120
1030 IF FT(H)>=FT(H+1) THEN 1080
1040 FT=FT(H):FT(H)=FT(H+1)
1050 FT(H+1)=FT:N$=N$(H):N$(H)=N$(H+1)
1060 N$(H+1)=N$
1070 T=1
1080 H=H+1
1090 IF H<PL THEN 1030
1100 IF T=1 THEN 1020:REM not refined
1110 GOTO 1300
1120 CLS
1130 PRINT
1140 PRINT "YOUR TOTAL POINTS, AFTER 2 ROUNDS=";
     TP(H)+WE(H)
1160 GOTO 1390
1190 REM tally first round points
1200 REM tally first round points
1210 IF TR>=2 THEN RETURN
1220 FOR I=1 TO PL
1230 WE(I)=TP(I):NEXT:RETURN
1240 REM final tally
1250 FOR I=1 TO PL
1260 FT(I)=WE(I)+TP(I)
1270 NEXT
1280 RETURN
1300 REM top winner(more than 1 player)
1310 CLS
1320 I=1
1330 PRINT "THE TOP WINNER: ";N$(I);" WITH ";
1340 PRINT FT(I);" POINTS."
1350 PRINT "FOLLOWED BY:"
1360 FOR I=2 TO PL:PRINT N$(I);" WITH ";
1370 PRINT FT(I);"POINTS."
```

```
1380 NEXT
1390 PRINT
1400 PRINT "END OF PROGRAM RUN."
1410 END
1500 REM instructions
1510 I$=" MIND INVASION "
1520 I=10:CLS
1530 FOR LX=1 TO LEN(I$)
1540 PRINT TAB(I) MID$(I$,LX,1);:SOUND 300,1
1550 FOR Y=1 TO 50:NEXT Y
1560 I=I+1:NEXT LX
1570 FOR Y=1 TO 1000:NEXT
1580 PRINT
1590 INPUT "NUMBER OF PLAYERS (1-10)";PL:SOUND 200,2
1600 IF PL<=0 OR PL>=11 THEN 1590
1610 PRINT "PLAY OUTLINE NECESSARY?";
1620 INPUT X$
1630 IF X$="N" OR X$="NO" THEN 2040
1640 PRINT
1650 PRINT "EACH OF YOU ";PL;" PLAYERS WILL
1660 PRINT "EXPERIENCE A NEW FORM OF MIND"
1670 PRINT "INVASION.  THE OBJECT OF THE GAME:"
1680 PRINT "KEEP PLAYING AS LONG AS POSSIBLE"
1690 PRINT "WITHOUT GETTING °CRUSHED'."
1700 PRINT "YOU, THE USER, WILL BE LOCATED AT"
1710 PRINT "THE CENTER OF THE VIDEO DISPLAY"
1720 PRINT "LOOKING LIKE THIS:";:COLOR 4:PRINT
     CHR$(2);:COLOR 2:PRINT " YOUR"
1730 PRINT "ONLY MOVEMENTS WILL BE STRAIGHT"
1740 PRINT "UP OR STRAIGHT DOWN, INDICATED"
1750 PRINT "BY THE <UP ARROW> AND <DOWN"
1760 PRINT "ARROW> KEYS."
1770 PRINT:GOSUB 2170
1780 PRINT "STRIKE A KEY TO CONTINUE"
1790 X$=INKEY$:IF X$=""THEN 1790
1800 PRINT
1810 PRINT "TO TAKE ADVANTAGE OF ANY POINTS"
1820 PRINT "YOU MUST BE CONSTANTLY MOVING,"
1830 PRINT "UP OR DOWN.  THE OUTER BOUNDARIES"
1840 PRINT "OF THE VIDEO WILL CONTAIN"
1850 PRINT "FLASHING ASTERISKS, FOR CONFUSION."
1860 PRINT "ALSO CONTAINED IN THOSE BOUNDARIES"
1870 PRINT "ARE BLOCKS, TWO OF WHICH,"
1880 PRINT "WITHOUT NOTICE, WILL MOVE VERY"
1890 PRINT "QUICKLY TOWARD YOUR LOCATION."
1900 PRINT "IF YOU ARE SUCCESSFUL IN"
1910 PRINT "MOVING OUT OF THE WAY, YOU"
```

```
1920 PRINT "CAN CONTINUE PLAYING.  IF, ON"
1925 GOSUB 2170:FOR LX=1 TO 5000:NEXT
1930 PRINT "THE OTHER HAND, YOU ARE CRUSHED"
1940 PRINT "PLAY WILL ADVANCE TO THE NEXT"
1950 PRINT "PLAYER.  THIS IS ALSO TRUE IF"
1960 PRINT "YOU COLLIDE WITH THE BLOCKS."
1970 FOR LX=1 TO 2000:NEXT
1980 GOSUB 2170:PRINT "STRIKE A KEY TO CONTINUE"
1985 X$=INKEY$:IF X$="" THEN 1985 ELSE PRINT
1990 PRINT "THE GAME WILL TERMINATE WHEN EACH"
2000 PRINT "PLAYER HAS PLAYED A TOTAL OF TWO"
2010 PRINT "ROUNDS.  THE PLAYER FINISHING WITH"
2020 PRINT "THE HIGHEST POINTS WILL BE THE"
2030 PRINT "WINNER."
2040 PRINT
2050 PRINT "NOW ENTER THE FIRST NAMES OF"
2060 PRINT "THE ";PL;" PLAYERS:"
2070 FOR I=1 TO PL:PRINT "PLAYER #";I;:INPUT N$(I):
     SOUND 300,1
2080 NEXT
2090 PZ=1
2100 PRINT N$(PZ);" WILL NOW PLAY.  STRIKE"
2105 IF PZ>=2 OR TR>=2 THEN 2150
2110 PRINT "A KEY WHEN READY:"
2120 X$=INKEY$:IF X$=""THEN 2120
2130 CLS
2140 RETURN
2150 PRINT "THE ENTER KEY WHEN READY";
2160 INPUT X$:TI=0:RETURN
2170 MIND$="MNO1CFC+F+DGD+G+":IF FL=1 THEN GOTO 2200
2180 PLAY "T60 XMIND$;T120 XMIND$;":FL=1
2190 RETURN
2200 PLAY "T240 L4XMIND$;L8 XMIND$;XMIND$;XMIND$;
     XMIND$;"
2210 RETURN
```

```
 1 CLEAR 700
 5 CLEAR 700
10 REM program title  KNIGHTS
15 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
   number generator
20 V=VAL(V$)
25 RANDOMIZE V
30 HORN$="MB T100 O2 L16E- L4B- L16E- L4B-"
35 KEY OFF:WIDTH 80:COLOR 1,11,14:CLS
40 PRINT TAB(30)">>>>KNIGHTS<<<<":PLAY "XHORN$;"
45  PRINT
50 PRINT "INSTRUCTIONS NEEDED";
55 INPUT T$
60 IF T$="N" OR T$="NO" THEN 205
65 PRINT "THERE ARE TEN KNIGHTS IN A CASTLE.
   THESE ARE KNIGHTS WHO HAVE DONE WRONG"
70 PRINT
75 PRINT "AND WERE IN A DUNGEON WAITING TO HAVE
   THEIR HEADS REMOVED."
80 PRINT
85 PRINT "THEY HAVE ESCAPED THE DUNGEON AND CAPTURED
   THE CASTLE.  THEY HAVE WITH THEM"
90 PRINT
95 PRINT "EIGHT HOSTAGES."
100 PRINT
105 PRINT "THE KING WILL NOT MEET THE CONDITIONS
    THEY HAVE SET FORTH, SO THEY VOW"
110 PRINT
115 PRINT "TO FIGHT TO THE DEATH"
120 PRINT:INPUT "PRESS <ENTER> TO CONTINUE",T$:PLAY
    "XHORN$;"
125 PRINT
130 PRINT "YOU ARE THE SHARPSHOOTER SELECTED BY THE
    KING TO TERMINATE "
135 PRINT
140 PRINT "THESE KNIGHTS.  YOU ARE EQUIPPED WITH A
    HIGH-POWER CROSSBOW."
145 PRINT
150 PRINT "ALONG THE CASTLE WALLS ARE FIVE WINDOWS.
    EACH TIME A FIGURE"
155 PRINT
160 PRINT "APPEARS AT A WINDOW YOU'LL HAVE THE
    OPTION TO FIRE (BY STRIKING"
165 PRINT
170 PRINT "THE SPACE BAR) OR NOT.  THE FIGURE THAT
```

```
    APPEARS AT A WINDOW"
175 PRINT
180 PRINT "COULD BE ONE OF THE HOSTAGES; BUT IF IT
    IS A KNIGHT AND YOU DON'T"
185 PRINT
190 PRINT "FIRE, YOU COULD BE HIT..."
195 PRINT
200 INPUT "PRESS <ENTER> TO BEGIN",T$:PLAY "XHORN$;"
205 CLS:K=10:HD=8:CR=10
210 SCREEN 1,0 'draw castle
215 COLOR 9,0
225 LINE (20,60)-(290,148),3,BF
230 FOR X=20 TO 280 STEP 20
235 LINE (X,50)-((X+10),60),3,BF
240 NEXT X
245 LINE (20,40)-(90,60),3,BF
250 LINE (220,40)-(290,60),3,BF
255 X=20
260 FOR Y=X TO (X+60) STEP 20
265 LINE (Y,30)-((Y+10),40),3,BF
270 NEXT Y
275 X=X+200:IF X<=220 THEN 260
280 FOR X=55 TO 255 STEP 200
285 LINE (X,10)-(X,40),3
290 NEXT X:PLAY "O3L8CL12D-E-D-L8CP16O2L16A-L4B-"
295 FOR X=56 TO 256 STEP 200     'draw flags
300 LINE (X,10)-(X,20),2
305 LINE -((X+20),15),2
310 LINE -(X,10),2
315 PAINT ((X+2),15),2
320 NEXT X
325 Z=70:Y=40:FOR X=4 TO 0 STEP -2      'draw and
    save windows
330 LINE ((Y-X),(Z-X))-((Y+20+X),(Z+40+X)),(X/2),BF
335 NEXT X
340 DIM WINDOW(340):GET(36,66)-(64,114),WINDOW
345 PI=3.14159:CIRCLE(50,110),7,3,-2*PI,-1*PI:PAINT
    (50,109),3    'draw and save figure
350 CIRCLE(50,96),7,3:PAINT(50,96),3
355 DIM FIGURE(340):GET(36,66)-(64,114),FIGURE
357 DIM HOLE(34):GET(290,160)-(300,170),HOLE
    'hole for arrows that miss
360 FOR X=36 TO 282 STEP 50      'window placement
365 PUT(X,66),WINDOW, PSET
370 NEXT X
590 REM get a window
600 W=INT(RND*5+1)
```

```
610 WI=W*50-14
615 BLANK$=STRING$(119,0)
620 REM pause before placing
630 PA=INT(RND*700+1):IF PA<=250 THEN 630
640 FOR PU=1 TO 700
650 IF PU=PA THEN 670
660 NEXT:GOTO 630
670 PUT(WI,66),FIGURE, PSET
680 REM sharpshooter to fire
690 FOR TI=1 TO 150
700 W$=INKEY$
710 IF W$<>""THEN 730
720 NEXT
730 REM knight or hostage
740 H1=INT(RND*5+1):H2=INT(RND*5+1):H3=INT(RND*5+1)
750 H=ABS(H1-H2-H3)
760 PUT(WI,66),WINDOW, PSET
762 IF H=W THEN 1240
765 IF HD<=0 THEN 840
770 IF H<=0 OR H>=5 AND TI<=99 THEN 790
780 GOTO 840
790 REM a hostage
800 IF TI>=112 THEN 880
810 SOUND 300,2:SOUND 100,3:LOCATE 20,1:PRINT
    "YOU HAVE JUST KILLED ONE"
820 PRINT "OF THE HOSTAGES, TURKEY!!"
830 GOTO 970
840 IF TI>=150 THEN 880
850 FOR SO=400 TO 1000 STEP 50:SOUND SO,1:NEXT:
    LOCATE 20,1:PRINT "GOOD SHOOTING, YOU HAVE"
860 PRINT "ELIMINATED ANOTHER KNIGHT!"
870 GOTO 940
880 IF H>=0 AND H<=2 THEN 1020
890 FOR SO=1000 TO 400 STEP -50:SOUND SO,1:NEXT:
    LOCATE 20,1:PRINT "THINK FASTER! YOU HAVE BEEN"
900 PRINT "HIT BY ONE OF THE KNIGHTS!!"
910 HK=HK+1
920 IF HK>=6 THEN 1060
930 GOTO 990
940 K=K-1:PRINT K;" KNIGHTS REMAIN."
950 IF K<=0 THEN 1120
960 GOTO 990
970 HD=HD-1
980 IF HD<=0 THEN 1180
990 FOR TX=1 TO 3000:NEXT
1000 REM clear print area
1010 LOCATE 20,1:PRINT BLANK$;:GOTO 14000U HAVE BEEN"
```

```
1020 IF HD<=0 THEN 890
1030 PLAY "XHORN$;":LOCATE 20,1:PRINT "GOOD THING
     YOU DID NOT FIRE;"
1040 PRINT "THAT WAS A HOSTAGE!!"
1050 GOTO 990
1060 REM hit enough
1080 PLAY "MB T60 O2DL8D.L16DL4DL8FEEDD.L16C+L4D"
     :LOCATE 20,1:PRINT "YOU HAVE BEEN HIT ";HK;
     " TIMES."
1090 PRINT "YOUR BODY CANNOT STAND"
1100 PRINT "ANY MORE.  YOU ARE DYING....."
1110 GOTO 1530
1120 REM knights killed
1140 LOCATE 20,1:PRINT "YOU HAVE DESTROYED ALL TEN"
1150 PRINT "KNIGHTS.  THE KING WILL BE"
1160 PRINT "PROUD OF YOU, MY FRIEND...."
1170 GOTO 1530
1180 REM hostages gone
1200 LOCATE 20,1:PRINT "YOU HAVE KILLED ALL EIGHT"
     IMES."
1210 PRINT "OF THE HOSTAGES, YOU TURKEY!!"
1220 PRINT "THE KING WILL HAVE YOUR HEAD!!"
1230 GOTO 990
1240 REM complete miss
1245 IF TI>=100 THEN 765
1250 XX=INT(RND*260+20)
1270 YY=INT(RND*38+110)
1290 PUT(XX,YY),HOLE,PSET
1310 LOCATE 20,1:PRINT "YOU'LL HAVE TO DO BETTER"
     :PLAY "T120 O1 B-D"
1320 PRINT "THAN THAT, YOU'VE MISSED COMPLETELY!!"
1330 GOTO 990
1340 REM crossbow rounds
1350 REM crossbow rounds
1360 PRINT
1370 PRINT "YOU HAVE ";CR;" ARROWS LEFT TO FIRE."
1380 RETURN
1390 REM enough arrows left
1400 REM enough arrows left
1410 IF TI<=99 THEN CR=CR-1
1420 GOSUB 1350:IF CR<=0 THEN 1440
1430 GOTO 590
1440 REM end
1450 FOR TX=1 TO 2000:NEXT
1460 CLS
1470 PRINT "YOU HAVE EXHAUSTED YOUR"
1480 PRINT "SUPPLY OF CROSSBOW ARROWS."
```

```
1490 IF K<>0 THEN 1510
1500 GOTO 1530
1510 PRINT "THERE ARE STILL ";K;" KNIGHTS "
1520 PRINT "IN THE CASTLE..."
1530 PRINT
1540 IF HD<=0 THEN 1560
1550 GOTO 1600
1560 PRINT "YOU ARE A BLUNDERING SHARP-"
1570 PRINT "SHOOTER, YOU IDIOT!!"
1580 PRINT "YOU HAVE HIT AND KILLED ALL"
1590 PRINT "EIGHT HOSTAGES!!!"
1600 PRINT:FOR T=1 TO 5000:NEXT
1610 PRINT "END OF RUN."
1615 SCREEN 0,0,0
1620 END
```

```
  5 CLEAR 100
 10 REM program title: INJECTION
 15 V$=RIGHT$(TIME$,2): 'use PC clock to seed
    random number generator
 20 V=VAL(V$)
 25 RANDOMIZE V
 30 COLOR 5,7,3:WIDTH 40:KEY OFF:CLS
 35 PRINT TAB(15)"> INJECTION <"
 40 PRINT
 45 INPUT "NUMBER TO PLAY (1-10)";U
 50 IF U<1 OR U>10 THEN 45
 55 PRINT "THE INITIALS OF PLAYERS:"
 60 FOR I=1 TO U:INPUT K$(I)
 65 NEXT:PRINT:PL=1:CLS
 70 PRINT "YOU ARE HEAD OF A LABORATORY"
 75 PRINT "IN WHICH YOUR COMPANY IS IN THE"
 80 PRINT "PROCESS OF INJECTING CUBES WITH"
 85 PRINT "A SOLVENT, TO DESTROY A COLONY"
 90 PRINT "OF BACTERIA WHICH CANNOT BE"
 95 PRINT "CONTROLLED.  THESE CUBES WILL"
100 PRINT "RACE ACROSS THE SCREEN FIVE AT A"
105 PRINT "TIME.  TO INJECT A CUBE, HOLD"
110 PRINT "THE <UP ARROW> KEY.  THE INJECTED"
115 PRINT "SOLVENT MUST STRIKE THE OPENING"
120 PRINT "AT THE BOTTOM OF THE CUBES."
125 PRINT
130 PRINT
135 INPUT "PRESS <ENTER> TO CONTINUE";L
140 CLS:PRINT
145 PRINT "EACH TIME A CUBE IS INJECTED,"
150 PRINT "THE NUMBER RACING ACROSS THE"
155 PRINT "SCREEN WILL DECREASE BY ONE, BUT"
160 PRINT "NO FEWER THAN THREE WILL RACE"
165 PRINT "ACROSS AT ONE TIME.  A TOTAL"
170 PRINT "OF TEN CUBES MUST BE INJECTED."
175 PRINT "IF AT ANY TIME YOU MISS AN"
180 PRINT "INJECTION, THE NUMBER OF CUBES WILL"
185 PRINT "INCREASE BY ONE.  THE INJECTION"
190 PRINT "MACHINE CAN BE MOVED LEFT OR"
195 PRINT "RIGHT, WITH THE LEFT AND RIGHT"
200 PRINT "ARROW KEYS.  GOOD LUCK."
205 PRINT
210 INPUT "PRESS ENTER TO BEGIN",L
215 CLS
220 REM the cubes
```

```
225 SCREEN 1,0:COLOR 0,1:JJ=0
230 DIM ALL(1992):DIM INJECTOR(804)
235 DIM BLANK(228):GET(0,0)-(30,28),BLANK
240 LINE(2,2)-(28,26),2,BF:
245 LINE(11,20)-(15,26),3,B:PAINT(12,21),3:DIM CUBE
    (228):GET(0,0)-(30,28),CUBE:PUT(0,0),BLANK,PSET
260 REM setting up graphics
270 FOR X=5 TO 205 STEP 50
280 PUT(X,60),CUBE,PSET
290 NEXT
300 GET(0,60)-(235,88),ALL:DIM SERUMBLANK(17):GET
    (164,138)-(166,151),SERUMBLANK
310 REM draw injection machine
317 LINE (165,146)-(165,138),1:DIM SERUM(17):GET
    (164,138)-(166,151),SERUM
320 LINE (150,169)-(180,199),3,BF:LINE(140,197)-
    (190,199),3,BF:LINE (150,169)-(1 65,159),1:LINE -
    (180,169),1:LINE -(150,169),1:PAINT(165,160),1,1
    :LINE (165,159)-(165,149),1
325 GET(133,149)-(197,199),INJECTOR
340 CLS:
410 JN=0:T=1:IN=10:E=0:CV=5:GOSUB 600
420 REM move cubes
430 X=0:INC=30:MINC=INC:M=X:J=0:JJ=0:P=1
440 PUT(X,60),CUBE,PSET
470 X=X+INC:J=J+1:SOUND 10*X+50*P,2:P=P+.1
480 IF JJ=1 THEN GOTO 520
490 IF J>=CV THEN 510
500 GOTO 440
510 REM blank out
520 PUT(M,60),BLANK,PSET
540 M=M+MINC
550 REM recycle
560 IF X>=259 OR X<=0 THEN INC=-INC:JJ=1:GOTO 580
570 IF M>=259 OR M<=0 THEN MINC=INC
580 GOTO 720
600 INJ=150:REM injector
610 PUT (INJ,149),INJECTOR,PSET:SOUND INJ*10+50,1
670 CA=0:QR=138:QC=INJ+31
690 LOCATE 1,1:PRINT "CUBES REMAINING:";
700 LOCATE 2,1:PRINT "CUBES ADDED:";
710 PRINT K$(PL):RETURN
720 REM injection/injector move
730 IF R=72 THEN 790
740 RR$=INKEY$:IF LEN(RR$)=2 THEN R$=MID$(RR$,2,1):
    R=ASC(R$)
750 IF R=72 THEN 790
```

```
760 IF R=75 THEN 900
770 IF R=77 THEN 960
780 GOTO 440
790 IF JN=1 THEN 810
800 QQ=INJ:QK=QC:JN=1
810 IF QR<=65 THEN V=1:GOTO 870
815 IF POINT(QC,QR-5)=0 THEN GOTO 830
820 IF POINT(QC,QR-5)=2 THEN V=1:PUT(QC,QR),
    SERUMBLANK,PSET:GOTO 870
825 IF POINT(QC,QR-5)=3 THEN PAINT(QC,QR-5),1,3:PUT
    (QC,QR),SERUMBLANK,PSET:GOTO 1060
830 QR=QR-5:PUT(QC,QR),SERUM,PSET:SOUND 10*(24-QR),1
840 GOTO 440
870 INJ=QQ+E:QC=INJ+31:QQ=0:E=0:QR=138:P=1
875 PUT(INJ,149),INJECTOR,PSET
880 IF V=1 THEN GOSUB 1140
890 R=0:JN=0:GOTO 440
900 REM to left
910 IF INJ<=0 THEN 440
930 INJ=INJ-T:QC=INJ+31
940 PUT(INJ,149),INJECTOR,PSET
950 GOTO 440
960 REM to right
970 IF INJ>=254 THEN 440
990 INJ=INJ+T:QC=INJ+31
1000 PUT(INJ,149),INJECTOR,PSET
1010 GOTO 440
1060 REM injected
1070 T=T+1:CV=CV-1:FOR B=1 TO 3:SOUND 2000,1:SOUND
     2250,1:NEXT B
1080 IF T>5 THEN T=1
1090 IF CV<=2 THEN CV=5
1100 IN=IN-1:LOCATE 1,20:PRINT IN;
1110 IF IN<=0 THEN 1250
1120 E=INT(RND*5):IF E+QQ>=256 THEN E=-(RND*5)
1130 GOTO 870
1140 REM cube added/injection missed
1150 IN=IN+1:FOR B=500 TO 600 STEP 20:SOUND B,1:
     NEXT B
1160 CA=CA+1:V=0
1170 T=T-1
1180 IF T<=0 THEN T=1
1190 LOCATE 1,20:PRINT IN;
1200 LOCATE 2,15:PRINT CA;
1210 RETURN
1250 REM player advance
1260 W(PL)=CA:PL=PL+1
```

```
1270 IF PL>U THEN 1310
1280 LOCATE 3,1:PRINT "STANDBY, ";K$(PL);"...";
1290 FOR YY=1 TO 1500:NEXT
1300 R=0:GOTO 410
1310 FOR YY=1 TO 900:NEXT:CLS
1320 PRINT "ALL PLAYERS HAVE HAD THEIR"
1330 PRINT "TURNS AT THE INJECTION MACHINE."
1340 PRINT "SCORES ARE BASED ON THE NUMBER"
1350 PRINT "OF CUBES THAT WHERE ADDED DURING"
1360 PRINT "THE INJECTION PROCESS."
1370 PRINT:FOR I=1 TO U
1380 PRINT "INJECTOR: ";K$(I)
1390 PRINT "HAD ";W(I);" CUBES ADDED."
1400 PRINT "FINAL SCORE EVALUATION: ";
1410 W=W(I)
1420 IF W>=0 AND W<=5 THEN 1470
1430 IF W>=5 AND W<=10 THEN 1480
1440 IF W>=10 AND W<=20 THEN 1490
1450 PRINT "TOO HORRIBLE TO MENTION."
1460 GOTO 1500
1470 PRINT "EXCELLENT.":GOTO 1500
1480 PRINT "FAIR.":GOTO 1500
1490 PRINT "POOR."
1500 FOR YY=1 TO 1200:NEXT YY:PRINT
1510 NEXT I
1520 PRINT
1530 PRINT "END OF INJECTION"
1535 SCREEN 0,0,0
1540 END
```

```
  1 REM program title: POINT A TO POINT B
  5 CLEAR 300
 10 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
    number generator
 15 V=VAL(V$)
 20 RANDOMIZE V:DIM SONG$(13)
 30 COLOR 14,10,14:CLS:KEY OFF:WIDTH 40:DEF
    SEG=&HB800
 40 PRINT TAB(15) "< FROM POINT A >":BEEP
 50 PRINT TAB(15) "< TO POINT B >":SOUND 500,3
 60 PRINT:DIM D$(50),S(50)
 70 BEEP:INPUT "INSTRUCTIONS (Y/N)";W$
 80 IF W$="N" OR W$="NO" THEN G=1
 90 BEEP:INPUT "NUMBER TO PLAY (1-10)";A
100 IF A<1 OR A>10 THEN 90
110 IF G=1 THEN G=0:GOTO 410
120 PLAY "MB L8 F E F A C":LOCATE 3,1:PRINT "ARE YOU
    CONFUSED BY THE MANY"
130 PRINT "DIFFERENT MAZE GAMES THAT ARE"
140 PRINT "CURRENTLY ON THE MARKET? HAVE"
150 PRINT "NO FEAR!! <FROM POINT A TO "
160 PRINT "POINT B> IS HERE!!"
170 PRINT:PRINT "ACTUALLY, THIS IS NOT YOUR"
180 PRINT "RUN-OF-THE-MILL MAZE GAME."
190 PRINT "IT IS ONE THAT WILL PROBABLY"
200 PRINT "CONFUSE YOU MORE THAN EVER!"
210 PRINT "YOU SEE, YOU WILL NOT BE"
220 PRINT "CONSTANTLY PRESSING DIFFERENT"
230 PRINT "KEYS FOR DIFFERENT DIRECTIONS;"
240 PRINT "INSTEAD, YOU'LL INPUT ALL"
250 PRINT:INPUT "PRESS ENTER TO CONTINUE",W
260 CLS
270 PLAY "MB L8 C C+ D E F":PRINT "OF THE
    INSTRUCTIONS BEFORE YOU"
280 PRINT "MOVE AT ALL: UP SO MANY"
290 PRINT "SPACES, RIGHT SO MANY SPACES,"
300 PRINT "LEFT, SO MANY, ETC., ETC."
310 PRINT "AFTER YOU HAVE ENTERED ALL"
320 PRINT "OF YOUR DIRECTION INSTRUCTIONS"
330 PRINT "YOU WILL INPUT <XX>.  AT THAT"
340 PRINT "POINT YOUR PIECE WILL BEGIN "
350 PRINT "TO MOVE FOLLOWING THE INSTRUCTIONS"
360 PRINT "YOU GAVE.  IF AT ANY POINT YOU "
370 PRINT "COLLIDE WITH ANYTHING OTHER"
380 PRINT "THAN A BLANK SPACE--END OF TURN..."
```

```
390 PRINT
400 INPUT "PRESS ENTER, AGAIN",W
410 CLS
420 REM players
430 REM play in order of entry
440 BEEP:PRINT "ENTER THE FIRST NAMES OF"
450 PRINT "THE ";A;"PLAYERS, PLEASE"
460 FOR I=1 TO A:INPUT A$(I):SOUND 100+25*I,3
470 NEXT:L=1:MX=0
480 PRINT
490 PRINT "TO BEGIN, ";A$(L);", PRESS ENTER";
500 INPUT W
510 REM  graphics and music
515 FOR I=1 TO 13:READ SONG$(I):NEXT
520 CLS:SNG=1:PLAY "XSONG$(SNG);":GOTO 580
560 LOCATE 1,1:PRINT STRING$(40,32):LOCATE 2,2:PRINT
    STRING$(36,219);
565 DC=INT(RND*6+25):DR=2
570 LOCATE DR,DC:PRINT STRING$(2,32);:RETURN
580 GOSUB 560:LOCATE 8,2:PRINT STRING$(9,219);
590 LOCATE 8,18:PRINT STRING$(9,219);:GOTO 620
600 LOCATE 8,22:PRINT STRING$(14,219)
605 DC=INT(RND*7+30):DR=6
610 LOCATE DR,DC:PRINT STRING$(2,32);:RETURN
620 GOSUB 700:M=10:M1=20:M2=6:M3=12:GOTO 670
630 SNG=SNG+1:IF SNG<=13 THEN PLAY "XSONG$(SNG);"
631 IF M2<=0 THEN M2=INT(RND*4+2)
632 IF M2>=23 THEN M2=INT(RND*10+13)
633 IF M1<=0 THEN M1=INT(RND*10+1)
634 IF M1>=40 THEN M1=38
635 FOR X=M TO M1:LOCATE M2,X:PRINT CHR$(219)
637 DR=8:DC=INT(RND*10+10):FR=11:FC=INT(RND*10+20)
638 LOCATE DR,DC:PRINT STRING$(10,219):LOCATE FR,FC:
    PRINT STRING$(10,219)
639 FOR ROW=12 TO 14:COL=INT(RND*25+1):GAP=INT(RND*
    15+1):LOCATE ROW,COL:PRINT STRING$(GAP,32):NEXT:
    LOCATE 8,COL:PRINT STRING$(GAP,32)
640 LOCATE M3,X:PRINT CHR$(219):NEXT:FOR X=M2+1 TO
    M3-2
650 LOCATE X,M:PRINT CHR$(219):LOCATE X,(M+1):PRINT
    CHR$(219):LOCATE X,(M1-1):PRINT CHR$(219)
660 LOCATE X,M1:PRINT CHR$(219):NEXT:RETURN
670 GOSUB 630
680 M=1:M1=5:M2=12:M3=16:GOSUB 630
690 M=6:M1=9:GOSUB 630
700 M=M1+1:M1=M1+22:M2=M3-2:M3=M3+6
710 GOSUB 630
```

```
 720 M=M1+1:M1=M1+9:M2=M2+6:M3=M3+6
 730 GOSUB 630
 740 M=1:M1=9:GOSUB 630:GOTO 770
 750 LOCATE 20,10:PRINT STRING$(25,219)
 752 LOCATE 12,30:PRINT STRING$(8,219)
 755 DC=INT(22*RND+10):DR=20
 758 FC=INT(6*RND+31):FR=12
 760 LOCATE DR,DC:PRINT STRING$(2,32);
 765 LOCATE FR,FC:PRINT STRING$(2,32);:RETURN
 770 GOSUB 750:M1=M1-3:M2=M3+2:M3=M3+6
 775 GOSUB 630
 780 M=M1+1:M1=M1+4
 790 GOSUB 630
 800 IF M>=38 THEN 880
 810 GOTO 780
 820 M=M-5:M1=M+3:M2=2:M3=10
 830 GOSUB 630
 840 M=M-4:M1=M+4:M2=M3+2:M3=M3+8
 850 GOSUB 630
 860 M=M-5:M1=M+1:M2=4:M3=6
 870 GOSUB 630:RETURN
 880 REM start/stop locations
 890 GOSUB 820:Y=21:X2=40
 900 X1=INT(RND*(X2-1)+2)
1000 IF G=1 THEN 1110
1010 COLOR 11:LOCATE 23,5:PRINT "THIS IS POINT A
     (START)."
1020 LOCATE Y,X1:PRINT CHR$(32):LOCATE Y,(X1-1):
     PRINT CHR$(32)
1030 FOR T=1 TO 50:NEXT
1040 COLOR 27:LOCATE Y,X1:PRINT CHR$(219)
1050 FOR T=1 TO 1000:NEXT
1060 COLOR 11:LOCATE Y,X1:PRINT CHR$(219)
1070 IF G=1 THEN 1150 ELSE JJ=X1
1080 G=1
1090 LOCATE 23,5:PRINT "LOOK AT THE TOP (POINT B)."
1100 Y=1:X2=30:GOTO 900
1110 LOCATE 23,5:PRINT STRING$(35,32)
1130 LOCATE 23,5:PRINT "THIS IS POINT B (STOP
     ZONE)."
1140 R=X1:GOTO 1020
1150 LOCATE 23,5:PRINT STRING$(35,32)
1160 GOTO 1210
1170 LOCATE 23,5:PRINT STRING$(35,32):LOCATE 23,5
1180 RETURN
1190 FOR T=1 TO 2000:NEXT
1200 GOSUB 1170:RETURN
```

```
1210 GOSUB 1170
1220 PRINT "YOU WILL NOW GIVE INSTRUCTIONS:
1230 GOSUB 1190:GOSUB 1170
1240 PRINT "TO ENTER THE STOP ZONE..."
1250 GOSUB 1190:GOSUB 1170
1260 PRINT "STOP ZONE BLOCK MUST BE HIT:
1270 GOSUB 1190:GOSUB 1170
1280 PRINT "HEAD-ON TO COMPLETE A RUN."
1290 GOSUB 1190:GOSUB 1170
1300 I=1
1310 INPUT "DIRECTION (L,R,U,D)",D$:SOUND 500,3
1320 IF D$="XX" THEN 1380
1330 D$(I)=D$:GOSUB 1170
1340 INPUT "FOR HOW MANY SPACES";S:BEEP
1350 S(I)=S:I=I+1:GOSUB 1170
1360 IF I>=51 THEN 1430
1370 GOSUB 1200:GOSUB 1170:GOTO 1310
1380 GOSUB 1170
1390 PRINT "THE COMPUTER WILL NOW CARRY"
1400 GOSUB 1190:GOSUB 1170
1410 PRINT "OUT ALL OF YOUR INSTRUCTIONS."
1420 I=I-1:H=I:I=1:X=JJ:Y=21:G=0
1425 GOSUB 1190:GOTO 1480
1430 GOSUB 1170
1440 PRINT "THAT IS THE LIMIT OF THE"
1450 GOSUB 1190:GOSUB 1170
1460 PRINT "INSTRUCTIONS YOU CAN ENTER."
1470 GOSUB 1190:GOTO 1380
1480 D$=D$(I):S=S(I):J=1:IF I<=12 THEN PLAY
     "XSONG$(I);"
1490 IF D$="L" THEN 1540
1500 IF D$="R" THEN 1580
1510 IF D$="U" THEN 1620
1520 IF D$="D" THEN 1660
1530 GOTO 1760  'bad direct
1540 IF PEEK((X-2)*2+(Y-1)*80)=219 THEN 1760
1550 X=X-1:LOCATE Y,X+1:PRINT CHR$(32)
1560 IF X<=0 THEN 1760
1570 GOTO 1690
1580 IF PEEK(X*2+(Y-1)*80)=219 THEN 1760
1590 X=X+1:LOCATE Y,(X-1):PRINT CHR$(32)
1600 IF X>=40 THEN 1760
1610 GOTO 1690
1620 IF PEEK((X-1)*2+(Y-2)*80)=219 THEN 1760
1630 Y=Y-1:LOCATE (Y+1),X:PRINT CHR$(32)
1640 IF Y<=0 THEN 1760
1650 GOTO 1690
```

```
1660 IF PEEK((X-1)*2+Y*80)=219 THEN 1760
1670 Y=Y+1:LOCATE (Y-1),X:PRINT CHR$(32)
1680 IF Y>=20 THEN 1760
1690 COLOR 11:LOCATE Y,X:PRINT CHR$(2):COLOR 14
1700 IF S>J THEN 1720
1710 GOTO 1730
1720 J=J+1:GOTO 1490
1730 I=I+1
1740 IF I>H THEN HH=1:GOTO 1760
1750 GOTO 1480
1760 GOSUB 1170:IF ABS(X-R)<=1 THEN 2060
1765 IF HH=1 THEN HH=0:GOTO 1900
1770 PRINT "SORRY...YOU HAVE ATTEMPTED"
1780 GOSUB 1190:GOSUB 1170
1790 PRINT "A SHORT-CUT THAT DIDN'T WORK."
1800 REM another play?
1810 GOSUB 1190
1820 GOTO 1950
1830 GOSUB 1190:GOSUB 1170
1840 PRINT "READY TO TRY AGAIN (Y/N)";
1850 INPUT W$
1860 IF W$="Y"THEN 1880
1870 GOTO 2300
1880 LOCATE Y,X:PRINT CHR$(32):LOCATE 1,R:PRINT
     CHR$(32)
1890 GOTO 1970
1900 GOSUB 1170
1910 PRINT "THAT WASN'T ENOUGH TO GET "
1920 GOSUB 1190:GOSUB 1170
1930 PRINT "YOU TO THE STOP ZONE, ";A$(L)
1940 GOTO 1800
1950 REM player advance
1960 IF L>=A THEN 2020 ELSE L=L+1
1970 GOSUB 1170
1980 PRINT A$(L);" WILL NOW ATTEMPT THE"
1990 GOSUB 1190:GOSUB 1170
2000 PRINT "VENTURE FROM POINT A TO B."
2005 LOCATE Y,X:PRINT CHR$(32):LOCATE 1,R:GOSUB 560
2010 GOSUB 600:GOSUB 750:M=35:M1=38
2015 GOSUB 1190:GOTO 880
2020 REM start over
2030 L=1:GOSUB 1170
2040 PRINT "ALL HAVE PLAYED..."
2050 GOSUB 2160:MX=0:GOTO 1830
2060 IF ABS(Y-1)>=1 THEN 1900
2070 GOSUB 1190:GOSUB 1170:REM complete
2080 PRINT "VERY WELL DONE, ";A$(L);"!!"
```

```
2090 GOSUB 1190:GOSUB 1170
2100 PRINT "YOU HAVE MADE A COMPLETE "
2110 GOSUB 1190:GOSUB 1170
2120 PRINT "PASSAGE TO POINT B..."
2130 GOSUB 1190
2140 M(L)=1:MX=1
2150 GOTO 1950
2160 REM players made to point b
2170 IF MX=0 THEN RETURN
2180 GOSUB 1190:GOSUB 1170
2190 PRINT "THE FOLLOWING HAVE EASILY MADE"
2200 GOSUB 1190:GOSUB 1170
2210 PRINT "IT TO POINT B...."
2220 GOSUB 1190:GOSUB 1170
2230 FOR U=1 TO A
2240 IF M(U)=1 THEN 2260
2250 NEXT:RETURN
2260 PRINT A$(U);"....."
2270 GOSUB 1190:GOSUB 1170
2280 M(U)=0:GOTO 2250
2290 REM end
2300 GOSUB 1190:GOSUB 1170
2310 PRINT "POINT A TO POINT B"
2320 GOSUB 1190:GOSUB 1170
2330 PRINT "TERMINATED...."
2340 GOSUB 1190:GOSUB 1170
2350 END
2360 DATA "MB T200 L1 O3 E D"
2365 DATA "MB T250 L8 C O2 B O3 C D C O2 G E F G A
     G E L4 G"
2370 DATA "MB MS L8 O3 C D L4 E E L8 E D C D L4 E
     D D"
2380 DATA "MB L8 O3 E D L8 C O2 B O3 C D C O2 G E F
     G A G E L4 G"
2390 DATA "MB MS L8 O3 C D L4 E G L8 G E C D L4 E D
     L2 C"
2400 DATA "MB MS O3 L8 E L4 G L8 E L4 G G L8 E L4 G
     L8 E L2 G"
2410 DATA "MB MS O3 L8 F L4 A L8 F L4 A A L8 F L4 A
     L8 F L2 A"
2420 DATA "MB MS L12 O3 G A B O4 L4 C C O3 G G E
     E D"
2425 DATA "MB MS L8 C D L4 E L8 G A G E C D L4 E
     D C"
2430 DATA "MB MS P4 O3 L4 C O2 L9 G G- G L4 G+ G"
2440 DATA "MB MS P4 O1 G O2 B O3 C"
2450 DATA "MB MS P4 O5 L4 C O4 L9 G G- G L4 G+ G"
2460 DATA "MB MS P4 O1 G O4 B O5 C"
```

```
  1 REM program title: INTERCEPT
  3 CLEAR 200
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
    number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 30 CLS:WIDTH 40:KEY OFF
 40  BEEP:PRINT TAB(15)"[[ INTERCEPT ]]"
 50 PRINT
 60 BEEP:INPUT "INSTRUCTIONS ";Q$
 70 IF Q$="N" OR Q$="NO" THEN 520
 80 PRINT
 90 PRINT "YOU ARE THE LEADER OF A SPECIAL"
 95 PRINT
100 PRINT "GROUP KNOWN AS [[ INTERCEPT ]]."
105 PRINT
110 PRINT "YOU WILL HAVE TO IDENTIFY AND
115 PRINT
120 PRINT "POSSIBLY DESTROY ANY ALIENS WHO"
125 PRINT
130 PRINT "VENTURE INTO THE STERILIZED DOME"
135 PRINT
140 PRINT "WHERE ALL OF YOUR INHABITANTS LIVE."
145 PRINT
150 PRINT "THE APPROACH TO THE DOME CONSISTS OF"
155 PRINT
160 PRINT "CROSSROADS. NEAR THE EDGE OF THE SCREEN"
165 PRINT
170 PRINT "ARE TWO LASER CANNONS.  THESE CANNONS"
175 PRINT
180 PRINT "WILL FIRE ONLY WHEN YOU HAVE ENTERED"
185 PRINT
190 PRINT "THE CORRECT KEY SEQUENCE: RIGHT ARROW,":
    GOSUB 2300:GOSUB 2330
193 PRINT
195 PRINT
197 INPUT "PRESS ENTER TO CONTINUE",Q$:CLS
200 PRINT "THEN LEFT ARROW TO PRIME, AND UP ARROW"
203 PRINT
205 PRINT "TO FIRE.":PRINT
210 PRINT "AT RANDOM TIMES, DIFFERENT FIGURES"
215 PRINT
220 PRINT "WILL APPEAR AT THE CENTER OF THE"
225 PRINT
230 PRINT "VERTICAL ARM OF THE CROSSROAD.  WHEN"
```

```
235 PRINT
240 PRINT "THE FIGURE IS AT THE CENTER, IN LINE"
245 PRINT
250 PRINT "WITH YOUR LASER, YOU MUST DECIDE "
255 PRINT
260 PRINT "WHETHER IT IS ALIEN OR FRIENDLY.  IF"
265 PRINT
270 PRINT "YOU BELIEVE THAT IT IS THE LATTER OF"
275 PRINT
280 PRINT "THE TWO, DO NOTHING--JUST LET IT PASS.:
285 PRINT
290 PRINT "IF YOU BELIEVE IT IS THE ALIEN, FIRE ":
    GOSUB 2310:GOSUB 2350
295 PRINT:PRINT
300 INPUT "PRESS ENTER TO CONTINUE:";Q$
310 CLS:PRINT "THE LASER BY PRESSING THE UP ARROW
    KEY."
315 PRINT
320 PRINT "REMEMBER, THE LASER CANNOT BE FIRED "
325 PRINT
330 PRINT "UNTILL YOU HAVE FIRST PRIMED IT BY "
335 PRINT
340 PRINT "PRESSING THE RIGHT, THEN LEFT ARROW KEYS.
    ":GOSUB 2310
345 PRINT
350 PRINT "YOUR COMMISSION WILL LAST 10 DAYS.  IF"
355 PRINT
360 PRINT "YOU DESTROY TEN FRIENDLY INTRUDERS "
365 PRINT
370 PRINT "BEFORE THAT TIME, YOUR COMMISSION"
375 PRINT
380 PRINT "WILL BE TERMINATED IMMEDIATELY."
385 PRINT:PRINT
390 INPUT "PRESS ENTER TO CONTINUE:",X$
400 CLS:PRINT "THE FINAL OBJECT:
405 PRINT
410 PRINT "TO DESTROY AS MANY ALIEN INTRUDERS"
415 PRINT
420 PRINT "AS YOU CAN BEFORE YOUR COMMISSION "
425 PRINT
430 PRINT "EXPIRES."
460 PRINT
510 PRINT
520 PRINT
530 IF Q$="N" OR Q$="NO" THEN GOSUB 2300
535 GOSUB 2350:INPUT "YOUR LAST NAME, PLEASE:",P$
540 CLS:COLOR 15,0,7
```

```
550 REM intercept crossroads
560 X$=STRING$(16,219)
570 AC=3:AR=4:J=1
580 LOCATE AR,AC:PRINT X$:LOCATE AR+5,AC:PRINT X$
590 LOCATE AR,AC+20:PRINT X$:LOCATE AR+5,AC+20:PRINT
    X$
600 AR=AR+1:J=J+1
610 IF J>=5 THEN 630
620 GOTO 580
630 AAR=8:AAC=37:A$=CHR$(219)+CHR$(16)
640 AA$=CHR$(17)+CHR$(219)
650 COLOR 11:LOCATE AR,AC:PRINT A$:LOCATE AAR,AAC:
    PRINT AA$
660 LOCATE 16,5:PRINT "LEADER: ";P$
670 LOCATE 17,5:PRINT "PRESENT CONDITION:"
680 LOCATE 18,5:PRINT "LASER STATUS:"
690 LOCATE 19,5:PRINT "DAYS TO GO ON COMMISSION:"
700 REM get figures
710 F$(1)=CHR$(1):F$(2)=CHR$(3):F$(3)=CHR$(5):F$(4)
    =CHR$(8):F$(5)=CHR$(15)
750 REM alien
760 A$(1)=CHR$(2):A$(2)=CHR$(4):A$(3)=CHR$(6):A$(4)
    =CHR$(14):A$(5)=CHR$(64)
790 REM double size
800 FOR I=1 TO 5
810 F$(I)=F$(I)+F$(I)
820 A$(I)=A$(I)+A$(I)
830 NEXT:KR1=AR:KC1=AC+2:KR2=AR:KC2=AC+18
840 REM days/alien/friendly
850 DA=10:AI=0:FI=0:AD=0:FD=0:DF=0
860 C$="QUIET   ":CC$="ALERT"
870 L$="CHARGING":LL$="READY   "
880 LOCATE 17,24:PRINT C$
890 LOCATE 18,19:PRINT L$;
900 LOCATE 19,31:PRINT DA;
910 REM start location
920 LR=13:LC=20
930 REM condition
940 FOR T=1 TO 500
945 IF INT(T/200)=T/200 THEN SOUND T/40,3:SOUND
    32767,3
950 I=INT(RND*5+1)
960 N=INT(RND*5+1)
970 IF N=I THEN 1000
980 NEXT T
990 GOTO 940
1000 REM alert / intruder
```

```
1005 IF T<=200 THEN T=T+1:GOTO 1005
1010 X=0:LF=1:XC=0:DS=0:YU=0
1020 LOCATE 17,24:PRINT STRING$(6,32);
1030 FOR T=1 TO 100:NEXT
1040 LOCATE 17,24:COLOR 28:PRINT CC$
1050 FOR TT=1 TO 500:NEXT
1060 SOUND 1000,3:SOUND 32767,3:X=X+1:IF X<=4 THEN
     1060
1070 COLOR 12:LOCATE 17,24:PRINT "INTRUDER";
1080 R=INT(RND*2+1)
1090 IF R=2 THEN 1110
1100 LOCATE LR,LC:COLOR N:PRINT F$(N);:GOTO 1115
1110 LOCATE LR,LC:COLOR N:PRINT A$(N);
1115 YU=YU+1
1120 IF E=0 THEN EE$=INKEY$:IF LEN(EE$)=2 THEN
     E$=MID$(EE$,2,1):E=ASC(E$)
1125 IF LF=4 THEN 1360
1130 MM=INT(RND*2+1)
1140 IF MM=1 AND YU<=70 THEN 1180
1150 IF LR<=1 THEN 1520
1160 LR=LR-1:LOCATE LR+1,LC:PRINT STRING$(2,32);
1170 IF LF<=6 GOTO 1200 ELSE 1360
1180 IF LR>=12 THEN MM=2:GOTO 1150
1190 LR=LR+1:LOCATE LR-1,LC:PRINT STRING$(2,32);
1200 IF E=77 AND LF=1 THEN 1240
1210  IF E=75 AND LF=2 THEN 1250
1220 IF E=72 AND LF=3 THEN 1260
1230 GOTO 1270
1240 LF=2:GOTO 1280
1250 LF=3:GOTO 1280
1260 LF=4:GOTO 1280
1270 E=0:GOTO 1090
1280 COLOR 3:LOCATE 23,5:PRINT "LASER KEY SEQUENCE:";
1290 PRINT LF-1:COLOR 15:SOUND 300,3
1300 IF LF=3 THEN 1320
1310 GOTO 1270
1320 COLOR 12:LOCATE 18,19:PRINT LL$;:COLOR 15
1330 GOTO 1270
1360 REM firing
1370 IF R=1 THEN 1390
1380 COLOR N:LOCATE LR,LC:PRINT A$(N);:COLOR 15:GOTO
     1400
1390 COLOR N:LOCATE LR,LC:PRINT F$(N);:COLOR 15
1400 COLOR 12:LOCATE AR,AC+2:PRINT STRING$(15,222);
1410 LOCATE AAR,AAC-15:PRINT STRING$(15,221);:COLOR
     15
1415 FOR TT=1 TO 30:SOUND 150,1:SOUND 32767,1
```

```
1417 NEXT
1420 DEF SEG=&HB800:IF PEEK(600)=32 THEN 1440
1430 GOTO 1650
1440 COLOR 11:LOCATE 18,19:PRINT L$:COLOR 15:LF=1
1450 COLOR 11:LOCATE 23,5:PRINT "INDIRECT HIT
     ";:COLOR 15
1460 LOCATE KR1,KC1:PRINT STRING$(16,32)
1470 LOCATE KR2,KC2:PRINT STRING$(16,32)
1480 FOR TT=1 TO 10:SOUND 450,1:SOUND 32767,1
1490 NEXT
1500 GOTO 1090
1520 REM intruder in dome
1525 IF YU<=50 THEN 1190
1530 LOCATE LR,LC:PRINT "  ";
1540 COLOR 12:LOCATE 23,5:PRINT "INTRUDER IS NOW IN
     DOME...":COLOR 15
1550 FOR TT=1 TO 15:SOUND 900,1:SOUND 32767,1
1560 NEXT:LOCATE AR,AC+2:PRINT STRING$(32,32);
1565 LOCATE 23,5:PRINT STRING$(25,32)
1570 FOR TT=1 TO 200:NEXT:SOUND 1750,3
1575 IF R=1 THEN 1590
1580 GOTO 1610
1590 COLOR 3:LOCATE 23,5:PRINT "INTRUDER:
     FRIENDLY....  ";:COLOR 15
1600 GOTO 1620
1610 COLOR 11:LOCATE 23,5:PRINT "INTRUDER: ALIEN...
     ":COLOR 15
1620 FOR TT=1 TO 500:NEXT
1630 IF XC<=3 THEN XC=XC+1:GOTO 1565
1640 GOTO 1670
1650 COLOR 3:LOCATE 23,5:PRINT "INTRUDER: DESTROYED
     ...    ":COLOR 15
1655 FOR TT=1 TO 10:SOUND (TT+10)*10,1:SOUND 32767,1
1656 NEXT
1660 DS=1:LOCATE LR,LC:PRINT "  ";:GOTO 1550
1670 REM variable count
1680 LOCATE 23,5:PRINT STRING$(28,32)
1690 IF DS=1 THEN 1810
1700 IF R=1 THEN 1750
1710 AI=AI+1
1720 LOCATE 20,5:PRINT "YOU NOW HAVE";AI;"ALIENS IN
     THE DOME."
1740 GOTO 1790
1750 FI=FI+1
1760 LOCATE 21,5:PRINT "YOU HAVE WISELY LET";FI;
     "FRIENDS PASS."
1790 REM continue if days>=1
```

```
1800 GOTO 1900
1810 REM destroyed
1820 IF R=1 THEN 1870
1830 AD=AD+1
1840 LOCATE 21,5:PRINT "YOU HAVE DESTROYED ";AD;"
     ALIENS.      "
1850 GOTO 1900
1870 FD=FD+1
1880 LOCATE 22,5:PRINT FD;" FRIENDS HAVE NOW BEEN
     DESTROYED."
1890 REM
1900 REM dx for days decrease
1910 REM if less than or equal to 75
1920 DX=INT(RND*100+1)
1930 IF DX<=75 THEN DA=DA-1:DF=DF+1
1940 FOR TT=1 TO 1200:NEXT
1950 LOCATE 23,5:PRINT STRING$(34,32)
1960 IF DA<=0 OR FD>=10 THEN 1980
1970 FOR TT=1 TO 10:SOUND 200,1:SOUND 32767,1
1975 NEXT:GOTO 860
1980 IF FD>=10 THEN 2010
1990 LOCATE 23,5:PRINT "YOUR COMMISSION HAS
     EXPIRED...     "
2000 GOTO 2020
2010 LOCATE 23,5:PRINT "YOUR COMMISSION
     TERMINATED....     "
2020 FOR TT=1 TO 2000:NEXT
2030 CLS
2040 PRINT "DURING YOUR ";DF;" DAY COMISSION"
2050 PRINT "YOUR RECORD IS AS FOLLOWS,";P$;":"
2070 PRINT "ALIENS ALLOWED TO PASS INTO DOME:";AI
2090 PRINT "ALIENS DESTROYED: ";AD
2100 PRINT "FRIENDLY INTRUDERS DESTROYED: ";FD
2110 PRINT
2120 IF FD>=10 THEN 2240
2130 PRINT "YOUR COMMISSION HAS BEEN RATED: "
2140 IF AD>=1 AND AD<=5 THEN 2170
2150 IF AD>=5 AND AD<=9 THEN 2190
2160 IF AD>=10 THEN 2210
2170 PRINT "IMPROPER:  POOR USE OF LASER."
2180 GOTO 2270
2190 PRINT "FAIR: LASER USED, BUT INSUFFICIENTLY."
2200 GOTO 2270
2210 PRINT "EXCELLENT: ALL JUDGEMENTS MADE"
2220 PRINT "WITH UTMOST CAUTION."
2230 GOTO 2270
2240 PRINT "YOU HAVE DESTROYED ";FD;" FRIENDLY"
```

```
2250 PRINT "INTRUDERS, ";P$;".  DO NOT ATTEMPT"
2260 PRINT "TO REAPPLY FOR THIS COMMISSION."
2270 PRINT
2280 PRINT "TERMINATION OF PROGRAM."
2290 END
2300 INTERCEPT$="O2 MS L12 B- A F ML L4 G"
2310 PLAY "MB MN O2 L2 C L4 F;XINTERCEPT$;MN L4G C;'
2320 RETURN
2330 PLAY "MB XINTERCEPT$;XINTERCEPT$;MS L12 A F A
     L1 C;"
2340 RETURN
2350 PLAY "MB XINTERCEPT$;XINTERCEPT$;P15 L12 MS
     GGG L 1 O3 C;"
2360 RETURN
```

Insomnia for the IBM PC

```
 10 REM program title:INSOMNIA
 20 CLEAR 200:COLOR 1,11,7:CLS:KEY OFF:WIDTH 40
 30 PRINT "INSOMNIA"
 40 PRINT
 50 PRINT "SO YOU ARE HAVING PROBLEMS"
 60 PRINT "SLEEPING.  JUST SIT BACK, KEEP"
 70 PRINT "YOUR EYES ON THE VIDEO AND"
 80 INPUT "PRESS ENTER";X:CLS
 90 A=20:W=0
100 PRINT:PRINT TAB(A)":";
110 PRINT TAB(A-1)":";TAB(A+1)":";
120 I=INT(RND*2+1)
130 IF I=1 THEN 160
140 IF A>=39 THEN 170
150 A=A+2:GOTO 180
160 IF A<=10 THEN 150
170 A=A-2
180 W=W+1:IF W=15 THEN 350
190 IF W=700 THEN 210
200 GOTO 100
210 IF W<=800 THEN 230
220 GOTO 250
230 PRINT
240 W=W+1:GOTO 210
250 PRINT "GETTING SLEEPY NOW????"
260 FOR I=1 TO W:NEXT
270 IF W<=814 THEN 290
280 GOTO 300
290 PRINT:W=W+1:GOTO 270
300 FOR X=1 TO 40
310 LOCATE 1,X:PRINT CHR$(219):LOCATE 23,X:PRINT
    CHR$(219):NEXT
320 COLOR 9:FOR X=1 TO 23
330 LOCATE X,1:PRINT CHR$(219):LOCATE X,40:PRINT
    CHR$(219):NEXT
340 FOR I=1 TO W:NEXT:W=0:GOTO 400
350 LOCATE 16,20
360 PRINT "KEEP EYES HERE!"
370 GOSUB 600
380 GOTO 190
400 L=38:AR=2:AC=2:XR=22:XC=6:Y=AR
410 A$=STRING$(L,L*4+27)
420 LOCATE AR,AC:PRINT A$;
430 LOCATE XR,XC:PRINT A$;
440 A$=STRING$(L,32)
```

```
450 LOCATE AR,AC:PRINT A$
460 LOCATE XR,XC:PRINT A$
470 AR=AR+1:XR=XR-1:IF AR=12 THEN 530
480 IF XR=Y THEN 500
490 GOTO 410
500 IF W<=50 THEN 520
510 GOSUB 600:GOTO 610
520 W=W+1:IF W=25 THEN GOSUB 580
525 GOTO 400
530 IF W>1 THEN 480
540 LOCATE AR,10
550 PRINT "STARE AT THIS LOCATION";
560 GOSUB 600
570 GOTO 480
580 A2=AR-6:IF AR<=0 THEN AR=1
585 A1=AC+25:IF A1>=40 THEN A1=40
590 PRINT "NOW FOLLOW THE BARS";
600 FOR I=1 TO 1000:NEXT:RETURN
610 CLS
620 PRINT "IF YOU ARE NOT GETTING SLEEPY"
630 PRINT "BY NOW, YOU'RE A HOPELESS"
640 PRINT "CASE!!!"
650 GOSUB 600:PRINT
660 PRINT "THIS IS THE LAST PHASE OF"
670 PRINT "INSOMNIA.  IF IT FAILS, RUN"
680 PRINT "THE ENTIRE PROGRAM ONCE"
690 PRINT "AGAIN....."
700 GOSUB 600:PRINT
710 PRINT "KEEP YOUR EYES ON THE MOVING"
720 PRINT "BLOCK, THINKING TO YOURSELF"
730 PRINT "'I AM GETTING SLEEPY....."
740 INPUT "PRESS ENTER ";X:W=0
750 COLOR 4,2:CLS:P$=STRING$(2,219):A$=STRING$(2,32)
    :PC=PC+30
760 Y=20:FOR X=6 TO 9
770 LOCATE X,Y:PRINT CHR$(219):NEXT
780 PR=X:PC=20
790 GOSUB 1100
800 FOR X=6 TO 9
810 LOCATE X,Y:PRINT CHR$(32):NEXT
820 IF Z=1 THEN 930
830 FOR X=6 TO 9
840 LOCATE X,Y:PRINT CHR$(219):Y=Y-1:NEXT
850 GOSUB 1120
860 PC=PC-6
870 GOSUB 1100
880 Y=20
```

```
890 FOR X=6 TO 9
900 LOCATE X,Y:PRINT CHR$(32):Y=Y-1:NEXT
910 GOSUB 1120:
920 Z=1:GOTO 760
930 GOSUB 1120
940 FOR X=6 TO 9
950 LOCATE X,Y:PRINT CHR$(219):Y=Y+1:NEXT
960 PC=PC+5:LOCATE PR,PC:PRINT P$
970 Y=20:FOR X=6 TO 9
980 LOCATE X,Y:PRINT CHR$(32)
990 Y=Y+1:NEXT
1000 GOSUB 1120
1010 W=W+1
1020 IF W=5 THEN 1050
1030 IF W=50 THEN 1150
1040 Z=0:GOTO 760
1050 FOR X=12 TO 27
1060 LOCATE 6,X:PRINT CHR$(219):LOCATE 11,X:PRINT
     CHR$(219):NEXT
1070 FOR X=6 TO 11
1080 LOCATE X,12:PRINT CHR$(219):LOCATE X,27:PRINT
     CHR$(219):NEXT
1090 GOTO 1030
1100 LOCATE PR,PC:PRINT P$;
1110 RETURN
1120 LOCATE PR,PC:PRINT A$;
1130 RETURN
1150 REM end of program
1155 LOCATE 6,20:PRINT P$
1160 GOTO 1160
```

```
  3 CLEAR 200:CLS:KEY OFF:WIDTH 80
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed random
    number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 30 PRINT TAB(25) "-:-: LAP-THE-TRACK :-:-"
 40 PRINT :DIM A$(15),A(15),LA(15),T(15)
 50 INPUT "INSTRUCTIONS";A$
 60 IF A$<>"Y" AND A$<>"YES" THEN 300
 70 PRINT "UP TO FIFTEEN PEOPLE BET ON FOUR MOVING
    SPOTS THAT ARE IN A FAST RACE"
 80 PRINT
 90 PRINT "TO THE FINISH.  A COMPLETE RACE CONSISTS
    OF TEN LAPS.  EACH PLAYER"
100 PRINT
110 PRINT "WILL BE ABLE TO CHOOSE A LANE (1-4) AND
    MAKE A BET OF ANY DESIRED "
120 PRINT
130 PRINT "AMOUNT ON THAT LANE.  IF THERE ARE NO
    WINNERS, THE COMPUTER WILL"
140 PRINT
150 PRINT "PLACE THE MONEY ON A LANE OF ITS CHOICE
    IN THE NEXT RACE.  IF THE"
160 PRINT
170 PRINT "COMPUTER'S SPOT HAPPENS TO WIN, IT WILL
    NOT TAKE THE MONEY, IT WILL"
180 PRINT
190 PRINT "JUST BET ON ANOTHER LANE.  SO, NO MATTER
    HOW YOU SLICE IT, SOMEONE"
200 PRINT
210 PRINT "IS BOUND TO WIN.  (YOU CAN USE REAL
    MONEY IN THIS GAME IF YOU WANT TO.)"
220 PRINT
300 PRINT
310 INPUT "HOW MANY WILL PLAY";A
320 IF A<1 OR A>15 THEN 320
330 PRINT "INITIALS, PLEASE";
340 FOR I=1 TO A
350 INPUT A$(I):NEXT
360 PRINT
370 PRINT "YOU WILL NOW SELECT A LANE."
380 I=1
390 PRINT A$(I);": LANE NUMBER";
400 INPUT LA(I)
410 IF LA(I)<1 OR LA(I)>4 THEN 390
```

```
420 PRINT "BET WHAT AMOUNT?";
430 INPUT A(I)
440 I=I+1
450 IF I>A THEN 480
460 PRINT
470 GOTO 390
480 CLS:GOSUB 1300:IF ZZ=1 THEN 530
490 PRINT "IF MORE THAN ONE PLAYER WINS (SELECTION
    OF SAME LANE), ALL MONEY"
500 PRINT "WILL BE DIVIDED AS EQUALLY AS POSSIBLE."
530 PRINT:IF CB=1 THEN 960
540 PRINT "PRESS ENTER TO START THE RACE:"
550 INPUT A$:COLOR 14,6,10:CLS
560 REM outer track
570 S$=STRING$(80,219):LOCATE 1,1:PRINT S$:LOCATE
    6,1:PRINT S$
580 LOCATE 11,1:PRINT S$:LOCATE 16,1:PRINT S$
590 FOR X=2 TO 5:LOCATE X,1:PRINT CHR$(219):LOCATE
    X,80:PRINT CHR$(219)
600 NEXT
610 FOR X=12 TO 15:LOCATE X,1:PRINT CHR$(219):
    LOCATE X,80:PRINT CHR$(219)
620 NEXT
630 REM spot locations
640 GOSUB 1200:GOSUB 1410:L=1:IR=7:IC=1
650 LR(L)=IR:LC(L)=IC:RR(L)=IR:RC(L)=IC:PR(L)=
    IR-5:PC(L)=IC+30:LP(L)=0
660 IR=IR+1:IF L>=4 THEN 680
670 L=L+1:GOTO 650
680 FOR I=1 TO 4
690 COLOR I:LOCATE LR(I),LC(I):PRINT CHR$(254):NEXT
700 REM start race
705 PLAY "MF MS O3 L9 C F A O4 L8 C L16 CC L8 C O3
    L8 A L16 AA L8 A F A F L4 C"
710 LP=0:X$=CHR$(219):PK=LR(1)+1
720 I=INT(RND*4+1):M=ABS(I-5)
730 IF LC(I)>80 THEN 1000 ELSE LOCATE LR(I),
    LC(I):PRINT " ";
740 IF ABS(LC(I)-RC(I))>=60 THEN 1000
750 LC(I)=LC(I)+INT(RND*15+1):IF LC(I)>=80 THEN
    GOTO 1000
760 COLOR I:LOCATE LR(I),LC(I):PRINT X$:SOUND I*50,1
770 IF M+1<>I AND M+1<>5 THEN 790
780 GOTO 820
790 IF LC(M+1)>80 THEN LC(M+1)=80
795 LOCATE LR(M+1),LC(M+1):PRINT " "
```

```
 800 LC(M+1)=LC(M+1)+INT(RND*4+1):IF LC(M+1)>=80
     THEN GOTO 1000
 810 COLOR (M+1):LOCATE LR(M+1),LC(M+1):PRINT
     X$:SOUND (M+1)*50,1
 820 IF M-1<>I AND M-1<>0 THEN 840
 830 GOTO 870
 840 IF LC(M-1)>80 THEN LC(M-1)=80
 845 LOCATE LR(M-1),LC(M-1):PRINT " ";
 850 LC(M-1)=LC(M-1)+INT(RND*4+1):IF LC(M-1)>=80
     THEN GOTO 1000
 860 COLOR (M-1):LOCATE LR(M-1),LC(M-1):PRINT X$;:
     SOUND(M-1)*50,1
 870 IF M+2<>5 AND M+2<>6 THEN 890
 880 GOTO 720
 890 IF LC(M+2)>80 THEN LC(M+2)=80
 895 LOCATE LR(M+2),LC(M+2):PRINT " "
 900 LC(M+2)=LC(M+2)+INT(RND*4+1):IF LC(M+2)>=80
     THEN GOTO 1000
 910 COLOR (M+2):LOCATE LR(M+2),LC(M+2):PRINT X$:
     SOUND(M+2)*50,1
 920 IF LC(M)>80 THEN LC(M)=80
 925 LOCATE LR(M),LC(M):PRINT " "
 930 LC(M)=LC(M)+INT(RND*4+1):IF LC(M)>=80 THEN
     GOTO 1000
 940 COLOR M:LOCATE LR(M),LC(M):PRINT X$;:SOUND
     M*50,1
 950 GOTO 720
 960 CB=0:COLOR 14:LOCATE 18,1:PRINT "THE COMPUTER
     WILL PLACE THE AMOUNT OF ";USING AA$;AM
 970 RF=INT(RND*4+1)
 980 RF=INT(RND*4+1):PRINT:PRINT "ON LANE ";RF;"."
 985 MN(RF)=MN(RF)+AM:AM=0
 990 GOTO 540
1000 IF LC(I)>80 THEN LC(I)=80
1005 LOCATE LR(I),LC(I):PRINT " ";
1010 LC(I)=RC(I)
1020 LP(I)=LP(I)+1:LOCATE PR(I),PC(I):COLOR 14:
     PRINT LP(I);
1030 IF LP(I)>=10 THEN 1500
1040 GOTO 720
1200 COLOR 14:LOCATE 2,2:PRINT "LAPS - LANE 1:";
1210 LOCATE 3,2:PRINT "LAPS - LANE 2:";
1220  LOCATE 4,2:PRINT "LAPS - LANE 3:";
1230 LOCATE 5,2:PRINT "LAPS - LANE 4:";
1240 REM amounts bet per lane
1250 LOCATE 12,2:PRINT "LANE 1.....";
1260 LOCATE 13,2:PRINT "LANE 2.....";
```

```
1270 LOCATE 14,2:PRINT "LANE 3.....";
1280 LOCATE 15,2:PRINT "LANE 4.....";
1290 RETURN
1300 REM amounts on lanes
1310 T=1
1320 FOR I=1 TO A
1330 IF LA(I)=T THEN 1350
1340 NEXT:GOTO 1370
1350 MN(T)=MN(T)+A(I)
1360 GOTO 1340
1370 T=T+1
1380 IF T>=5 THEN 1400
1390 GOTO 1320
1400 RETURN
1410 REM print amounts
1420 AA$="$$#,###.##"
1430 I=1:FR1=12:FC1=13
1440 COLOR 14:LOCATE FR1,FC1:PRINT USING AA$;MN(I)
1450 I=I+1:FR1=FR1+1
1460 IF I>=5 THEN RETURN
1470 GOTO 1440
1500 REM finish
1510 WW=1
1520 COLOR 30:LOCATE PR(I),PC(I)+10:PRINT "WINNER"
     ;:FOR T=1 TO 3:SOUND 1000,1:SOUND 800,1:NEXT
1530 FOR TU=1 TO 1000
1540 NEXT:COLOR 14
1560 IF WW>=6 THEN 1580
1570 WW=WW+1:GOTO 1520
1580 REM player lane select
1590 WW=0
1600 FOR T=1 TO A
1610 IF LA(T)=I THEN 1630
1620 NEXT:GOTO 1650
1630 WW=WW+1:T(WW)=T
1640 GOTO 1620
1650 FOR I=1 TO 4:AM=AM+MN(I):MN(I)=0
1660 NEXT:IF WW=0 THEN 1760
1670 COLOR 14:LOCATE 12,30:PRINT "WINNER(S):";
1680 WXR=13:WXC=30
1690 FOR I=1 TO WW
1700  LOCATE WXR,WXC:PRINT A$(T(I));
1710 IF WW>1 THEN PRINT ",";
1720 WXC=WXC+5:NEXT
1725 IF WW=1 THEN 1790
1730 LOCATE 14,30:PRINT "EACH WINNER RECEIVES:";
1740 LOCATE 15,30:PRINT USING AA$;AM/WW;
```

```
1750 AM=0:GOTO 1820
1760 LOCATE PR(1),PC(1)+13:
1770 PRINT "** NO WINNERS **    ";
1780 GOTO 1870
1790 LOCATE 14,30:PRINT,:
1800 PRINT "YOU WILL RECEIVE:";
1810 GOTO 1740
1820 REM another
1830 LOCATE 7,1:PRINT,;
1840 INPUT "ANOTHER RACE (Y/N)";A$
1850 IF A$="N" THEN 2000
1860 COLOR 7,0,0:CLS:ZZ=1:GOTO 360
1870 FOR TU=1 TO 1200:NEXT
1880 COLOR 14:LOCATE 19,1
1890 PRINT "IF ANOTHER RACE IS HELD, THE";
1910 PRINT "COMPUTER WILL BET THE MONEY";
1930 PRINT "ON A LANE IT SELECTS.";
1940 CB=1:GOTO 1820
2000 REM end
2010 CLS
2020 PRINT "FAREWELL 'TIL NEXT RACE."
2030 END
```

```
  1 REM program title: BE PREPARED (PART I)
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed
    random number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 20 WIDTH 40:KEY OFF
 25 COLOR 15,11,10:CLS
 30 PRINT "BE PREPARED PART I)
 40 PRINT
 50 PRINT "IF YOU HAVEN'T DONE SO, READ AND"
 60 PRINT "FOLLOW ALL INSTRUCTIONS FOR PLAY."
 70 PRINT "YOU MUST UNDERSTAND EVERYTHING."
 80 PRINT
 90 INPUT "INSTRUCTIONS";Q$
100 IF MID$(Q$,2,1)<>"E" THEN 400
110 PRINT
120 PRINT "THIS GAME TAKES PLACE IN A"
130 PRINT " DISTANT OUTER SPACE"
140 PRINT "LOCATION.  YOU WILL SEE FOUR"
150 PRINT "TUBES WITH THE TOP CLOSED AND"
160 PRINT "THE BOTTOM OPEN FOR ENTRY.  THESE"
170 PRINT "ARE THE PORTS.  YOUR VESSEL,"
180 PRINT "REPRESENTED BY AN ASTERISK, WILL"
190 PRINT "BE LOCATED BELOW THE TUBES.  IT"
200 PRINT "WILL MOVE CONTINUOUSLY FROM LEFT"
210 PRINT "TO RIGHT.  ONE OF THE NUMERAL "
220 PRINT "KEYS 1 AND 2 WILL PROPEL YOUR"
230 PRINT "CRAFT INTO A TUBE, BUT YOU MUST"
250 PRINT
260 INPUT "PRESS ENTER TO CONTINUE:",X$
270 CLS
280 PRINT "GUESS WHICH KEY WILL DO THE JOB."
290 PRINT "ONLY ONE KEY WILL WORK EACH TURN"
300 PRINT "AND THE COMPUTER WILL INCREASE"
310 PRINT "THE NUMBER OF KEYS YOU MUST "
320 PRINT "CHOOSE BETWEEN EACH CYCLE (I.E.,"
330 PRINT "1 TO 3, THEN 1 TO 4, ETC.)"
340 PRINT "YOU HAVE TWENTY VESSELS TO BRING"
345 PRINT "SAFELY TO PORT.  EACH VESSEL"
350 PRINT "HAS A LIMITED AMOUNT OF FUEL."
360 PRINT "IF YOU UNDERSTAND ALL INSTRUC-"
370 PRINT "TIONS, PRESS ENTER TO PORT"
380 PRINT "THE FIRST VESSEL:";
390 INPUT X$
400 CLS
```

```
410 REM vessels/begin value
420 V=20:DS=0:DK=0:VL=2
430 REM tubes
440 X=3:Y=13
450 FOR T=Y TO 1 STEP -1
460 LOCATE T,X:PRINT CHR$(219):LOCATE T,(X+5):
    PRINT CHR$(219):NEXT
470 FOR T=X TO X+5
480 LOCATE 1,T:NEXT
490 REM do four
500 X=X+10
510 IF X>=43 THEN 530
520 GOTO 450
530 REM bottom of ports
540 W=1:X=3
550 FOR T=W TO X
560 LOCATE Y,T:PRINT CHR$(219):NEXT:IF P=1 THEN 600
570 W=X+6:X=X+10
580 IF W>=39 THEN X=40:P=1
590 GOTO 550
600 REM fuel
610 F=200
620 F$="FUEL REMAINING "
630 XR=16:XC=10
640 FOR I=1 TO LEN(F$)
650 LOCATE XR,XC:PRINT MID$(F$,I,1);
660 XC=XC+1:GOSUB 1500
670 NEXT:IF G=1 THEN RETURN
680 QC=XC:QR=XR
690 LOCATE QR,QC:PRINT F
700 REM start locale
710 LR=15:LC=1:V$="*":ZX=2550:PK=1
720 M=LC+39:P=INT(RND*VL+1):R=0
730 LOCATE LR,LC:PRINT V$;
740 REM by inkey$
750 X$=INKEY$
760 IF VAL(X$)=P THEN 900
770 FOR T=1 TO 5:NEXT
780 LOCATE LR,LC:PRINT " ";
790 IF R=1 THEN 810
800 LC=LC+1:GOTO 840
810 LC=LC-1
820 IF LC<=M THEN 720
830 GOTO 850
840 IF LC>=M THEN M=LC-39:R=1:VL=VL+1
850 F=F-1:IF VL>=9 THEN VL=9
860 IF F<=0 THEN 1200
```

```
 870 LOCATE QR,QC:PRINT F;:LOCATE QR,QC+5:PRINT
     "TONS";
 880 IF INT(F/10)=F/10 THEN 1540
 890 GOTO 730
 900 REM port?
 910 J=1:ZX=500:PK=0
 920 LOCATE LR,LC:PRINT " ";
 930 LR=LR-1:IF LR<=1 THEN 1000
 940 LOCATE LR,LC:PRINT V$;
 950 J=J+1:GOTO 1540
 960 DEF SEG=&HB800:IF PEEK((LR-2)*80+2*(LC-1))
     <>32 THEN 990
 970 GOTO 920
 990 IF J<=2 THEN 1030
1000 F$=" VESSEL  DOCKED  SAFELY "
1010 G=1:GOSUB 630
1020 DK=DK+1:GOTO 1070
1030 F$="V E S S E L  D E S T R O Y E D"
1040 FOR T=1 TO 500:NEXT:LOCATE LR,LC:PRINT " ";
1050 G=1:XR=20:XC=5:GOSUB 640
1060 DS=DS+1
1070 GOSUB 1600
1080 F$=STRING$(32,32):XR=20:XC=5
1090 GOSUB 640
1100 V=V-1:G=0
1110 IF V<=0 THEN 1300
1120 VL=VL+1
1130 IF VL>=9 THEN VL=9
1140 GOTO 600
1200 REM fuel exhausted?
1210 F$="F U E L  E X H A U S T E D"
1220 G=1:XR=20:XC=5
1230 GOSUB 640
1240 FOR T=1 TO 500:NEXT
1250 GOTO 1030
1300 REM all vessels in
1310 G=1:XX=1
1320 F$="A L L  V E S S E L S  I N    "
1330 XR=16:XC=5:GOSUB 640:GOSUB 1600:XR=XX:XC=5:CLS
1340 F$="YOU HAVE DOCKED"
1350 GOSUB 640:PRINT DK;:F$="VESSELS"
1355 XC=XC+4
1360 GOSUB 640:XR=XX+1:XX=XX+1:XC=5
1370 F$="YOU HAVE DESTROYED"
1380 GOSUB 640:PRINT DS;:F$="VESSELS"
1385 XC=XC+3
1390 GOSUB 640:XR=XX+1:XX=XX+1:XC=5
```

```
1400 F$="END OF MISSION..."
1410 GOSUB 640
1420 END
1500 REM sound routine
1510 SOUND 10*(ASC(MID$(F$,I,1))+10),1:SOUND 32767,1
1530 RETURN
1540 REM more sound
1550 SOUND ZX,1:SOUND 32767,1
1560 IF PK=1 THEN 1580
1570 GOTO 960
1580 IF F<=100 THEN ZX=ZX-100
1590 GOTO 890
1600 REM for message loop
1610 FOR T=1 TO 1200:NEXT
1620 RETURN
```

```
 10 REM program title:THE COURSE
 20 KEY OFF:COLOR 7,2,3:CLS:WIDTH 40:PRINT TAB(10);
 30 PRINT ">>> THE COURSE <<<"
 40 PRINT
 50 INPUT "REQUIRE INSTRUCTIONS";X$
 60 IF X$="N" OR X$="NO" THEN 210
 70 PRINT
 80 PRINT "THE COURSE WILL REQUIRE THAT YOU"
 90 PRINT "USE TOP NOTCH REFLEXES TO CONTROL"
100 PRINT "YOUR VEHICLE TO A DISTANCE OF"
110 PRINT "10 LAPS. THE COURSE WILL BE BOTH"
120 PRINT "STRAIGHT AND WINDING.  YOU WILL"
130 PRINT "CONTROL YOUR VEHICLE WITH THE"
140 PRINT "LEFT AND RIGHT ARROW KEYS."
150 PRINT "THE SLIGHTEST CONTACT TO THE"
160 PRINT "EDGE OF THE TRACK WILL CAUSE"
170 PRINT "AN ACCIDENT, AND A HIT"
180 PRINT "ADDED TO YOUR SCORE."
190 PRINT:PRINT:PRINT "STRIKE A KEY..."
200 X$=INKEY$:IF X$="" THEN 200
210 CLS:T=1:AT=1
220 INPUT "FIRST NAME OF DRIVER";D$
225 IF D$="" THEN PRINT:GOTO 220
230 PRINT D$;" STRIKE A KEY TO BEGIN..."
240 X$=INKEY$:IF X$="" THEN 240
245 IF T>=2 THEN CLS:GOTO 260
250 CLS:DIM CX(26),CY(26)
260 LOCATE 1,1:PRINT "DRIVER: ";D$;:GOSUB 1000
265 CC=0:LP=0:LL=0:HI=0:REM the course
270 CX=1940:CY=CX+18:J=219:C=1
275 C1=CX+6:C2=C1+2:REM the vehicle
280 KY=0:IF CX<=1 OR C1<=1 THEN 400
285 IF CC=1 THEN 420
290 DEF SEG=&HB800:POKE CX,J:POKE CY,J
295 CX(C)=CX:CY(C)=CY:REM to blank old course
300 CX=CX-80:CY=CY-80
305 C=C+1
310 DD=2*INT(RND*2+1)
315 KY$=INKEY$:IF LEN(KY$)=2 THEN KY=ASC(MID$
    (KY$,2,1))
320 IF DD=2 THEN 340
330 CX=CX+DD-2:CY=CY+DD-2:GOTO 360
340 DD=-DD
350 CX=CX+DD:CY=CY+DD
360 REM check for movement of vehicle
```

```
370 IF KY=75 THEN 500
380 IF KY=77 THEN 530
390 GOTO 280
400 CC=1:LP=LP+1
405 IF (LP/10)=.6 THEN LL=LL+1:LP=1
410 LOCATE 2,1:PRINT "LAPS=";LL;
415 IF LL>=10 THEN 700 ELSE GOTO 270
420 DEF SEG=&HB800:POKE CX(C),32:POKE CY(C),32
430 POKE C1,32:POKE C2,32
440 IF C2<=156 THEN 290
450 C1=C1-80:C2=C2-80
460 POKE C1,204:POKE C2,185:POKE C1+1,36:POKE
    C2+1,36
470 CL=PEEK(C1-2):CR=PEEK(C2+2)
480 IF ABS(CL+CR)<>64 THEN 600
485 GOTO 290
500 REM vehicle movement (left)
505 DEF SEG=&HB800:POKE C1,32:POKE C2,32:POKE
    C1+1,39:POKE C2+1,39
510 C1=C1-2:C2=C2-2
520 GOTO 280
530 REM vehicle movement (right)
535 DEF SEG=&HB800:POKE C1,32:POKE C2,32:POKE
    C1+1,39:POKE C2+1,39
540 IF C2<=84 THEN 280
550 C1=C1+2:C2=C2+2
560 GOTO 280
600 REM hits to vehicle
610 HI=HI+1
620 LOCATE 3,1:PRINT "HITS=";HI;
630 GOTO 290
700 REM finish
710 LOCATE 4,1:PRINT STRING$(40,32)
720 IF HI=0 THEN 800
730 PRINT "YOU DIDN'T MAKE THE";LL;"LAPS"
740 PRINT "WITHOUT ERROR.  YOU TOOK A"
750 PRINT "TOTAL OF";HI;"HITS...."
760 PRINT "TRY AGAIN, ";D$;
770 INPUT X$
780 IF X$="N" OR X$="NO" THEN 1100
790 AT=AT+1:PRINT:GOTO 830
800 PRINT "NOT BAD, NO HITS AT ALL"
810 PRINT "THINK YOU MIGHT WANT TO"
820 GOTO 760
830 T(T)=HI:REM best score
840 I=1:Z=0
850 TRON:IF T(I)<=T(I+1)THEN 8800K
```

```
860 YS=T(I):T(I)=T(I+1):T(I+1)=YS
870 Z=1
880 I=I+1:IF I>=T THEN 900
890 GOTO 850
900 IF Z=1 THEN 840
910 BS=T(1)
920 T=T+1:IF T>=5 THEN BS=T(1):T=3
930 PRINT
940 PRINT:GOTO 230
1000 REM message
1010 IF T<=2 THEN 1040
1020 LOCATE 1,28:PRINT "BEST SCORE: ";BS;
1030 LOCATE 2,28:PRINT "ATTEMPTS: ";AT;
1040 RETURN
1100 REM end
1110 CLS
1120 PRINT "END OF PROGRAM RUN..."
1130 END
```

```
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed
    random number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 20 REM program title <TRAPPED>
 25 CLS:KEY OFF:CLEAR 100:DIM AA(12),PI(12),I$(12)
    :WIDTH 40
 30 COLOR 1,0:PRINT TAB(15)CHR$(174);:COLOR 20,
    0:PRINT "TRAPPED";:COLOR 1,0:PRINT CHR$(175)
 40 COLOR 2,0:PRINT:INPUT "LIST INSTRUCTIONS";W$
 50 IF W$="Y" OR W$="YES" THEN GOSUB 3500
 60 CLS:GOSUB 3300:TU=1
 70 QQ=1:MM=0:CR=0
 90 S1=440:S2=S1+50:CC=0:BR=0:TP=10000!:TR=TP
100 REM border graphics
110 X=1920:Y=X+78:C=219:U=0
120 DEF SEG=&HB800:FOR I=X TO Y STEP 2:POKE I,
    222:POKE (I+1),2
130 POKE I-1920,222:POKE (I-1919),2:NEXT
140 FOR I=2 TO 22
150 POKE (I*80),222:POKE ((I*80)+1),2:POKE
    (I*80+78),222:POKE ((I*80)+79),2:NEXT :GOSUB 3450
160 COLOR 9,0   'rem inner graphics
170 IN=X-160
180 R=IN+2*INT(RND*39+1):R1=R:L=2*INT(RND*2+1)
190 FOR I=IN TO R STEP 2:POKE I,C
195 IF L=1 AND U<=3 THEN POKE I-80,C
198 NEXT
200 R2=2*INT(RND*7+1):R3=R2+2*INT(RND*10+1):GOTO 500
210 FOR I=R1+R2 TO R1+R3 STEP 2
220 POKE I,C:NEXT
230 GOTO 560
240 U=U+1
250 IF U<>5 THEN 180
260 REM graphics from right side
270 FOR J=1 TO INT(RND*10+1)
280 FOR I=0 TO 4
290 IF R(I)=0 THEN 310
300 POKE R(I),C:R(I)=R(I)-2
310 NEXT I,J
320 REM start position
330 P=2*INT(RND*842+79)
350 IF PEEK(P)<>32 OR PEEK(P+4)<>32 THEN 330
355 IF (PEEK(P+80)+PEEK(P-80))=438 THEN 330
360 CP=P+4:POKE P,220:POKE (P+1),14:POKE CP,223:POKE
    (CP+1),14
```

```
370 REM movement of computer piece
380 S3=210:S4=880
390 IF PEEK(P-80)=32 THEN A=1:GOTO 440
395 SOUND S1,3:SOUND S2,3:
400 IF PEEK(P+80)=32 THEN A=2:GOTO 440
410 IF PEEK(P-4)=32 THEN A=3:GOTO 440
420 IF PEEK(P+4)=32 THEN A=4:GOTO 440
430 GOTO 810
440 POKE P,32:IF PEEK(CP)=32 THEN 1700
442 FOR H=1 TO 100:X$=INKEY$:IF LEN(X$)=2 THEN 900:
    NEXT
445 GOTO 1100
450 ON A GOTO 460,470,480,490
460 P=P-80:POKE P,220:POKE P+1,12:MM=0:GOTO 710
470 P=P+80:POKE P,220:POKE P+1,12:MM=0:GOTO 400
480 P=P-4:POKE P,220:POKE P+1,12:GOTO 760
490 P=P+4:POKE P,220:POKE P+1,12:GOTO 420
500 REM within limits (board graphics)
510 IF ABS((R1+R3)-IN)>=X THEN 200
520 REM gap too small
530 IF ABS((R1+R2)-R)<=4 THEN 200
540 IF R3<=9 THEN 200
550 GOTO 210
560 REM poke more to right
570 IF ABS((IN-78)-(R1+R3))>=10 THEN 600
580 X=X-240:IN=X-160:POKE IN,C
590 GOTO 240
600 R4=2*INT(RND*8+1):IF R4<=6 THEN 600
610 R4=R4+R1+R3
620 R5=R4+2*INT(RND*10+1)
630 IF ABS(R5-R4)<=2 THEN 620
640 IF ABS(R5-IN)>=62 THEN 680
650 FOR I=R4 TO R5 STEP 2:POKE I,C
660 IF L=1 AND U<>4 THEN POKE (I-80),C
670 NEXT
680 REM variables (right side graphics)
690 FF=IN-76:R(U)=FF:N(U)=ABS(FF-R5)
700 GOTO 580
710 REM down/up-search left/right
720 IF PEEK(P-4)=32 THEN 410
730 IF PEEK(P+4)=32 THEN 420
740 IF A=1 THEN 390
750 GOTO 400
760 REM left/right search down/up
770 IF PEEK(P-80)=32 THEN 390
780 IF PEEK(P+80)=32 THEN 400
790 IF A=3 THEN 410
800 GOTO 420
```

```
810 REM trapped(counter)
815 A(QQ)=A
820 IF A(QQ)=A(QQ-1) THEN 880
830 IF MM>=8 THEN 1240
840 IF MM>=4 THEN S3=S3+10:CR=CR+10
860 QQ=QQ+1:IF QQ>=3 THEN QQ=1
870 GOTO 390
880 IF PEEK(P+4)=223 OR PEEK(P-4)=223 THEN MM=MM+1
890 GOTO 830
900 REM user piece/move per key
910 FOR H=1 TO 100:XX$=INKEY$:IF LEN(XX$)=2 THEN
    X$=MID$(XX$,2,1):NEXT
920 IF ASC(X$)=80 THEN 970
930 IF ASC(X$)=72 THEN 990
940 IF ASC(X$)=77 THEN 1010
950 IF ASC(X$)=75 THEN 1030
960 CC=0:GOTO 450
970 IF PEEK(CP+80)<>32 THEN 960
980 POKE CP,32:CP=CP+80:GOTO 1050
990 IF PEEK(CP-80)<>32 THEN 960
1000 POKE CP,32:CP=CP-80:GOTO 1050
1010 IF PEEK(CP+4)<>32 THEN 960
1020 POKE CP,32:CP=CP+4:GOTO 1050
1030 IF PEEK(CP-4)<>32 THEN 960
1040 POKE CP,32:CP=CP-4
1050 POKE CP,223:POKE CP+1,14:TP=TP-1
1060 SOUND S1,3:SOUND S2,3
1070 REM users moves(counter)
1080 CC=CC+1:IF CC>=10 THEN CC=0:GOTO 450
1090 GOTO 900
1100 REM sound/computer piece
1110 REM charge rate (diminish by plus)
1120 SOUND S1,3:SOUND S4,3
1130 CR=CR+1
1140 IF CR>=100 THEN GOSUB 1450:GOTO 1180
1150 IF CR>=99 THEN S4=S4+10
1170 GOTO 450
1180 REM recharge computer piece
1190 POKE P,220:POKE P+1,12:FOR C=1 TO ((S4*.2)
    -100)STEP 50
1200 SOUND 100*C,3:SOUND (S2-100),3
1210 NEXT:CR=0:S4=880:POKE P,32
1220 GOSUB 1510:GOSUB 3450:GOTO 450
1240 REM trapped?
1250 REM check charge rate first
1260 IF CR>=925 THEN 1280
1270 GOTO 1290
```

```
1280 CR=1000:GOTO 1370
1290 REM trapped computer piece
1300 L=0
1310 POKE P,32:FOR TI=1 TO 100:NEXT
1320 POKE R,220:POKE R+1,12:FOR TI=1 TO 50
1330 SOUND S1:SOUND S2
1340 NEXT
1350 L=L+1:IF L<=2 THEN 1310
1360 GOTO 1600
1370 GOSUB 1450:POKE P,32:MM=0
1380 P=P+6
1390 GOTO 450
1450 REM message
1460 MG=20:M$="RECHARGING"
1470 FOR I=1 TO LEN(M$)
1480 POKE MG,ASC(MID$(M$,I,1)):POKE MG+1,132
1490 MG=MG+2:NEXT
1500 POKE MG,32:RETURN
1510 MG=20
1520 FOR I=1 TO LEN(M$)
1530 POKE MG,222:POKE MG+1,2
1540 MG=MG+2:NEXT
1550 POKE MG,222:RETURN
1600 REM message (trapped)
1610 MG=20
1620 M$="**TRAPPED**"
1630 GOSUB 1470
1640 GOTO 1910
1700 REM user "burned"
1710 MG=20
1720 M$="* USER BURNED *"
1730 GOSUB 1470
1740 REM leave user position as-is
1750 L=0:MM=0
1760 POKE CP,32:FOR TI=1 TO 100:NEXT
1770 POKE CP,223:POKE CP+1,14:FOR TI=1 TO 5
1780 SOUND S1,3:SOUND (S1-180),3
1790 NEXT:IF L<=1 THEN L=L+1:GOTO 1760
1800 BR=BR+1:GOSUB 1510:GOSUB 3450
1810 IF BR>=6 THEN GOSUB 1830:GOTO 1870
1820 GOTO 445
1830 L=0
1840 FOR I=1 TO 45 STEP 9
1850 SOUND S2,3:SOUND (I*100),3
1860 NEXT:CLS:RETURN
1870 PRINT
1880 COLOR 2,0:PRINT:PRINT "YOU'VE BEEN BURNED ";
     BR;" TIMES"
```

```
1890 PRINT "SUCKER!!!";:TP=0
1900 PRINT "YOU'RE OUT OF THE GAME...";:GOTO 2000
1910 REM trapped
1920 GOSUB 1830
1930 PRINT "YA GOT ME TRAPPED, TURKEY!!!"
1940 PRINT "YOUR TOTAL POINTS NOW EQUAL ";
1950 PRINT TP
1960 PRINT "YOU MADE A TOTAL OF ";TR-TP;" MOVES"
1970 PRINT "BEFORE TRAPPING ME....."
1980 FOR H=1 TO 2:SOUND S1,3:SOUND (S1*2),3:NEXT
2000 REM player advance
2010 PI(TU)=TP
2020 IF TU=N THEN 2100
2030 TU=TU+1
2040 PRINT "PRESS ENTER WHEN READY, ";I$(TU);
2050 INPUT W$:CLS
2060 GOTO 70
2070 REM all played
2110 REM sort numbers/players
2120 IF N=1 THEN 2220
2130 F=0:AA=1
2140 IF PI(AA)>=PI(AA+1) THEN 2190
2150 P=PI(AA):PI(AA)=PI(AA+1)
2160 PI(AA+1)=P:I$=I$(AA)
2170 I$(AA)=I$(AA+1):I$(AA+1)=I$
2180 F=1
2190 IF AA<N THEN AA=AA+1:GOTO 2140
2200 IF F=1 THEN 2130
2210 GOTO 2260
2220 PRINT "YOU'VE PLAYED BY YOURSELF, AGAINST"
2230 PRINT "THE COMPUTER.  YOU WERE BURNED ";BR
2240 PRINT "TIMES.  FINAL TRAPPING SCORE: ";PI(TU)
2250 GOTO 2330
2260 I=1
2270 PRINT "BEST TRAPPER.....";I$(I)
2280 PRINT "TOTAL POINTS=";PI(I)
2290 PRINT "FOLLOWING ";I$(I)
2300 FOR I=2 TO N:PRINT I$(I)
2310 PRINT " WITH A SCORE OF: ";
2320 PRINT PI(I)
2330 FOR TI=1 TO 1000:NEXT TI,I:PRINT
2340 PRINT "END OF PROGRAM RUN...."
2350 FOR I=1 TO 100:SOUND S1,3:SOUND ((S1-50)+
     (I*10)),3
2360 NEXT:PRINT
2370 PRINT "UNLESS ALL WANT TO PLAY AGAIN???"
2380 PRINT "IF SO, ENTER <RUN> (THE <F1> KEY)."
```

```
2390 PRINT:FOR TI=1 TO 1500:NEXT
2400 PRINT "END"
2410 END
3300 REM player numbers/names
3310 PRINT "ENTER THE NUMBER OF PLAYERS:";
3320 INPUT N
3330 IF N<=0 OR N>=11 THEN 3310
3340 PRINT "NOW ENTER THE ";N;" PLAYERS'"
3350 PRINT "FIRST NAMES"
3360 FOR I=1 TO N
3370 INPUT I$(I):NEXT
3380 PRINT
3390 PRINT "CURRENT PLAYER'S NAME WILL  BE"
3400 PRINT "PRINTED AT THE BOTTOM OF"
3410 PRINT "GAME BOARD."
3420 FOR TI=1 TO 1200:NEXT
3430 CLS
3440 RETURN
3450 REM player
3460  DEF SEG=&HB800:MG=1920:MF=MG
3470 FOR I=1 TO LEN(I$(TU))
3480 POKE MG,ASC(MID$(I$(TU),I,1)):POKE MG+1,142:
     MG=MG+2
3485 NEXT:POKE MG,32:FOR H=1 TO 500:NEXT H
3490 MG=MF:FOR I=1 TO LEN(I$(TU)):POKE MG+1,14
    :MG=MG+2:NEXT I:POKE MG,32:RETURN
3500 REM instructions
3510 CLS:COLOR 2,0:PRINT "UP TO 10 PLAYERS CAN TRY"
3520 PRINT "TO OUTWIT THE COMPUTER AND
3530 PRINT "MANEUVER IT'S PIECE INTO A SITUATION"
3540 PRINT "WHERE IT WILL BECOME TRAPPED."
3550 PRINT "THE COMPUTER'S PIECE WILL BE A"
3560 PRINT "BLOCK LIKE THIS: ";:COLOR 12,0:PRINT
     CHR$(220);:COLOR 2,0:PRINT " YOURS"
3570 PRINT "WILL LOOK LIKE THIS: ";:COLOR 14,0:
     PRINT CHR$(223);:COLOR 2,0
3580 PRINT "THE COMPUTER WILL MOVE"
3590 PRINT "ITS PIECE ALL AROUND THE GAME"
3600 PRINT "BOARD, CHANGING DIRECTION WHEN IT"
3610 PRINT "SENSES AN OBSTACLE.  YOU WILL"
3620 PRINT "FOLLOW THE COMPUTER'S PIECE OR"
3630 PRINT "MANEUVER IT TO A SECTION"
3640 PRINT "OF THE GAME BOARD WHERE IT WILL BE"
3650 PRINT:PRINT "PRESS ENTER TO CONTINUE";:INPUT
     W$:CLS
3660 PRINT "TRAPPED, MEANING THAT IT CAN ONLY MOVE"
3670 PRINT "LEFT AND RIGHT.  THE COMPUTER'S PIECE"
```

```
3680 PRINT "WILL CONTAIN A 'CHARGE'.  IF YOU"
3690 PRINT "SIMULTANEOUSLY LAND ON THE SAME SPOT"
3700 PRINT "YOU'LL GET 'BURNED'.  "
3710 PRINT "SIX BURNS AND YOU'RE"
3720 PRINT "OUT OF THE GAME.  YOU WILL MOVE"
3730 PRINT "YOUR PIECE USING ALL FOUR ARROW"
3740 PRINT "KEYS.  WHEN YOU MOVE, THE COMPUTER"
3750 PRINT "WILL BLANK IT'S PIECE.  YOU"
3760 PRINT "WILL BE ALLOWED ONLY 10 CON-"
3770 PRINT "SECUTIVE MOVES AT ONE TIME."
3780 PRINT "ANY TIME YOU WISH TO MOVE YOUR"
3790 PRINT "PIECE, HOLD DOWN ANY ARROW KEY"
3800 PRINT "AND PRESS THE <SPACE BAR>."
3810 PRINT:INPUT "PRESS ENTER TO CONTINUE",W$
3820 CLS:PRINT "POINTS WILL BE DETERMINED BY HOW"
3830 PRINT "MANY MOVES IT TAKES THE USER"
3840 PRINT "TO TRAP THE COMPUTER'S PIECE."
3850 PRINT "REMEMBER, TO TRAP THE COMPUTER'S"
3860 PRINT "PIECE, ONLY ALLOW IT TO MOVE"
3870 PRINT "LEFT AND RIGHT."
3880 PRINT:INPUT "PRESS ENTER TO START",W$
3890 CLS:RETURN
```

Rainy Daze for the IBM PC

```
  1 REM program title: RAINY DAZE
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed
    random number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 20 CLS:WIDTH 40:KEY OFF
 30 INPUT "INSTRUCTIONS";A$
 40 INPUT "PLAYERS";P:IF P=0 THEN 40
 50 PRINT "THE";P;"PLAYERS'INITIALS:"
 60 IF P>=11 THEN 20 ELSE FOR I=1 TO P
 70 INPUT N$(I):NEXT
 80 IF A$<>"Y" AND A$<>"YES" THEN 340
 90 REM instructions
100 PRINT "THIS PROGRAM IS DESIGNED TO"
110 PRINT "HELP YOU THROUGH YOUR WORST"
120 PRINT "DAYS, BY MAKING THEM WORSE!!"
130 PRINT "ACTUALLY, THIS PROGRAM WILL"
140 PRINT "TEST YOUR ABILITY TO FOLLOW"
150 PRINT "NUMBERS, SOMETIMES IN SEQUENCE."
160 PRINT "THE NUMERALS 1 THROUGH 660 WILL"
170 PRINT "RACE ACROSS AND FILL THE VIDEO."
180 PRINT "WHEN A NUMERAL APPEARS THAT IS"
190 PRINT "NOT SUPPOSED TO BE THERE, HOLD"
200 PRINT "THE <CTRL> KEY FOR A SHORT"
210 PRINT "TIME UNTIL THAT NUMERAL IS"
220 PRINT "BLOCKED OUT."
230 PRINT
240 PRINT
250 INPUT "ENTER";X$
260 CLS
270 PRINT "POINTS WILL BE GIVEN ON THE BASIS"
280 PRINT "OF HOW MANY OUT-OF-SEQUENCE NUMERALS"
290 PRINT "YOU HAVE DETECTED AT THE END OF"
300 PRINT "THE COUNT, 660."
310 PRINT "REMEMBER TO KEEP GOOD EYE CONTACT "
320 PRINT "WITH THE SCREEN...":PRINT
340 I=1
350 IF P=1 THEN 390
360 PRINT N$(I);", YOU WILL NOW PLAY:"
370 INPUT "PRESS ENTER, PLEASE:",X$
380 GOTO 410
390  PRINT "WHEN READY, ";N$(P);", PRESS"
400  INPUT "THE ENTER KEY:",X$
410 CLS:N=0:N1=0:TT=500:DEF SEG=&H40:POKE &H17,0
420 AR=1:AC=1:AB=AC:M=30:L=2:G=0
```

```
430 DEF SEG=&H40:K=PEEK(&H17):GOTO 550
440 LOCATE AR,AC:PRINT N;
450 FOR T=1 TO TT+TT
460 NEXT:AC=AC+5
470 IF AC>=M THEN 490
480 GOTO 430
490 AR=L:AC=AB:L=L+1:TT=TT-15:G=0
500 IF L>=24 THEN 520
510 GOTO 430
520 FOR T=1 TO 1000:NEXT
530 CLS
540 GOTO 420
550 REM change or leave
560 IF K=4 THEN NJ=1:GOTO 690
570 N=N+1
580 N1=N1+1:IF N1<=2 THEN 610
590 J=INT(RND*2):IF G=1 THEN J=0:N=N1
600 IF J=1 THEN N=N+1:G=1
610 IF K=0 AND ABS(N-N1)<>0 THEN Y=Y+1
620 IF NJ=1 THEN NJ=0:GOTO 640
630 IF N1>=661 THEN 800 ELSE GOTO 440
640 N1=N1+1
645 IF AC-5<=0 THEN LOCATE AR-1,40-AC-5:PRINT
    STRING$(5,143);:GOTO 660
650 LOCATE AR,AC-5:PRINT STRING$(5,143);
660 REM pause before cont.
670 FOR T=1 TO 200:NEXT:K=0
680 GOTO 440
690 IF ABS(N=N1)=0 THEN 710
700 Y=Y-1:GOTO 720
710 Y=Y+1
720 DEF SEG=&HB800:A1=PEEK((AR-1)*80+2*(AC-7))
730 A2=PEEK((AR-1)*80+2*(AC-9))
740 IF A1=143 AND A2=143 THEN 760
750 GOTO 620
760 CLS:Y=Y+2
770 PRINT "QUIT HOLDING DOWN THE <CTRL>"
780 PRINT "KEY!! LET GO, YOU MANIAC!"
790 FOR T=1 TO 1200:NEXT:CLS:GOTO 620
800 REM player advance
810 YY(I)=Y:Y=0:CLS
820 I=I+1
830 IF I>P THEN N1=N1-1:GOTO 850
840 GOTO 360
850 REM final
860 PRINT "PLAYERS AND THEIR POINTS"
870 PRINT "ARE AS FOLLOWS:"
```

```
880 FOR I=1 TO P
890 PRINT N$(I);" IDENTIFIED ";
900 IF YY(I)=0 THEN PRINT "ALL NUMERALS.":GOTO 930
910 IF YY(I)<0 THEN YY(I)=-YY(I)
920 PRINT N1-YY(I);"NUMERALS."
930 NEXT
940 PRINT
950 PRINT "END OF PROGRAM."
960 END
```

```
  1 REM program title:general store
  3 CLEAR 500
  5 V$=RIGHT$(TIME$,2): 'use PC clock to seed
    random number generator
 10 V=VAL(V$)
 15 RANDOMIZE V
 20 DEF SEG=&HB800:WIDTH 40:CLS:PRINT TAB(10);
 30 PRINT "GENERAL STORE":R$="NEXT "
 40 DIM N$(15),I$(31),I(31)
 50 DIM L(31),PR(31),TP(15):REM instructions
 60 PRINT "YOUR INSTRUCTIONS ARE SIMPLE AND"
 70 PRINT "STRAIGHT-FORWARD:  I WILL GIVE"
 80 PRINT "YOU A LIST OF DIFFERENT ITEMS"
 90 PRINT "AND THEIR PRICES.  YOU IN TURN"
100 PRINT "WILL ENTER THE TOTAL PRICE OF ALL THE"
110 PRINT "ITEMS. THAT IS, IF I SAY $2.50,"
120 PRINT "MILK, AND THEN SAY $0.80 BREAD"
130 PRINT "YOU MUST THEN ENTER THE TOTAL"
140 PRINT "OF $2.50 + $0.80 ($3.30).  ALL"
150 PRINT "OF YOUR ENTRIES WILL BE THROUGH"
160 PRINT "THE COMPUTER'S 'INKEY$' FUNCTION."
170 PRINT "DON'T PRESS <ENTER> AND DON'T ENTER"
180 PRINT "THE DOLLAR SIGN.":PRINT:PRINT "STRIKE
    A KEY TO CONTINUE"
190 GOSUB 3000:CLS:G$="##,###"
200 PRINT "ENTER ONLY THE NUMBERS AND THE"
210 PRINT "DECIMALS.  AFTER 10 ITEMS WE WILL"
220 PRINT "TAKE A SHORT BREAK, THEN I WILL"
230 PRINT "GIVE YOU 10 MORE.  WE WILL STOP"
240 PRINT "ONLY AFTER 30 ITEMS HAVE BEEN"
250 PRINT "TOTALLED.  YOUR POINTS WILL BE"
260 PRINT "DETERMINED BY THE ACCURACY OF YOUR"
270 PRINT "TOTAL.  IF CORRECT, THE TOTAL WILL BE"
280 PRINT "YOUR POINTS.  IF INCORRECT, ALL ENTRIES"
290 PRINT "MISSED WILL BE SUBTRACTED FROM YOUR TO-"
300 PRINT "TAL.  YOU WILL HAVE A TIME LIMIT IN"
310 PRINT "WHICH TO ENTER THE TOTAL, SO LAY"
320 PRINT "DOWN THE PENCIL AND THE CALCULATOR!"
340 REM begin
350 PRINT "STRIKE A KEY TO BEGIN"
360 GOSUB 3000:CLS
370 INPUT "HOW MANY PLAYERS (1-10)";P
380 IF P<1 OR P>10 THEN 370
390 PRINT "ENTER";P;"PLAYERS' FIRST NAMES:"
400 REM names
```

```
410 AP=220:FOR I=1 TO P
420 INPUT N$(I):N(I)=I:NEXT:N=1:AR=3:AC=30
425 AQ=AP+80:IF P=1 THEN Y=1:N=2:GOTO 470
430 PRINT "I WILL SELECT EACH PLAYER IN"
440 PRINT "A RANDOM FASHION.  THERE WILL"
450 PRINT "BE NO SET ORDER OF PLAYERS."
460 GOSUB 2800:PRINT     'player
470 IF EE=1 THEN 1530    'final
480 Q$=N$(Y):TA=0:JI=0
490 PRINT Q$;" WILL NOW PLAY:"
500 PRINT "PRESS THE ENTER KEY TO BEGIN";
510 INPUT X$:PRINT
520 CLS:PRINT "HOW FAST IS YOUR ADDITION,"
530 PRINT Q$;"???"
540 REM read all items/prices
550 FOR I=1 TO 30:READ I$(I):L(I)=I:NEXT
560 FOR I=1 TO 30:READ PR(I):NEXT:GOTO 2500
570 D=I-1:DD=(D/3):U=1:D$="$##.##":F$="$"
580 REM select items randomly
590 IF U>DD THEN PRINT:PRINT:GOTO 1100
600 X=INT(RND*30+1)
610 IF L(X)=0 THEN 600  'used
620 T=X:PR=PR(T):T$=I$(T)
625 IF U=1 THEN YU=PR
630 TA=TA+PR     'total amount/current item
640 TA$=STR$(TA)         'total to string
645 IF U=1 THEN GOSUB 2900
650 IF U=1 THEN CLS ELSE  IF U>1 THEN 670
660 PRINT TAB(1)"PRICE";TAB(12)"ITEMS";
665 PRINT TAB(30)"PRICES";
670 PRINT TAB(1) USING D$;PR;
680 PRINT TAB(12);T$
690 IF U<=1 THEN 710
700 RR=1        'continue print/entries
710 L(X)=0:U=U+1:KK=0:NE=1
720 IF RR<>1 THEN 580
725 LOCATE AR,AC:PRINT F$
730 X$=INKEY$:IF KK<=15 AND X$<>"" THEN 1040
735 IF X$="." THEN 880
740 IF X$="" THEN 760
750 GOTO 800
760 KK=KK+1      'time expired?
770 IF KK>=400 THEN W$="":GOTO 1000
780 POKE AP,220:GOTO 730
800 REM entry
810 POKE AP,ASC(X$):TY(NE)=VAL(X$)
820 AP=AP+2      'poke location
```

```
 830 NE=NE+1      'number entry
 840 IF NE=LEN(TA$) THEN 860
 850 GOTO 730
 860 GOTO 910     'all entries in
 880 REM decimal
 890 POKE AP,ASC(X$)
 900 TY(NE)=1000:GOTO 820
 910 REM compile entries
 920 V=NE-1:IF V=9 THEN V=10
 930 FOR I=1 TO V
 940 IF TY(I)=1000 THEN 970
 950 W$=W$+STR$(TY(I))
 960 TY(I)=0:NEXT:GOTO 990
 970 W$=W$+"."
 980 GOTO 960
 990 W=VAL(W$):W$=STR$(W)
1000 IF W$<>TA$ THEN PI=PI+PR:JI=JI+1
1005 IF U=3 AND JI<>0 THEN PI=PI+PR:JI=JI+1
1008 IF PEEK(AP)=220 THEN POKE AP,32
1010 POKE AP+2,32:W$="":AR=AR+1
1020 AP=AQ:AQ=AQ+80
1030 GOTO 590      'another item
1040 REM      slow down a little
1050 POKE AP,32
1060 GOTO 730
1100 REM points/player advance
1110 IF PI<>0 THEN 1200
1120 FOR I=1 TO LEN(TA$)
1130 IF MID$(TA$,I,1)="." THEN 1150
1140 K$=K$+MID$(TA$,I,1)
1150 NEXT
1160 K=VAL(K$)
1170 GOSUB 1430
1180 TP(Y)=TP(Y)+K
1190 GOTO 1320
1200 REM missed items
1210 TA$=STR$(PI):FOR I=1 TO LEN(TA$)
1220 IF MID$(TA$,I,1)="." THEN 1240
1230 K$=K$+MID$(TA$,I,1)
1240 NEXT:K=VAL(K$):GOSUB 1430
1250 TP(Y)=TP(Y)-K
1260 PRINT "YOU HAVE MISSED TOTALS ON ";JI
1270 PRINT "ITEMS, ";Q$;". THESE ITEMS CAME"
1280 PRINT "TO A TOTAL OF $";TA$;".  YOU ARE"
1290 PRINT "MINUS THAT AMOUNT.  YOUR SCORE"
1300 PRINT "IS NOW ";USING G$;TP(Y)
1310 GOTO 1350
```

```
1320 PRINT "YOU HAVE SUCCESSFULLY ENTERED"
1330 PRINT "ALL TOTALS, ";Q$;". YOUR POINTS"
1340 PRINT "ARE NOW: ";USING G$;TP(Y)
1350 K=0:JI=0:PI=0:TA$="":K$=""
1360 U=1:D1=D1+10:IF D1=20 THEN R$="LAST"
1370 AP=220:AQ=AP+80:AR=3:AC=30:RR=0:PRINT
1380 IF D1>=30 THEN 1500
1390 PRINT "TAKE A SHORT BREAK, THEN WE'LL"
1400 PRINT "ATTEMPT THE ";R$;" SET OF ITEMS."
1410 FOR I=1 TO 5000:NEXT:PRINT
1420 GOTO 580
1430 REM amount times 10
1440 IF MID$(TA$,LEN(TA$)-1,1)="." THEN 1480
1450 IF LEN(TA$)=2 OR LEN(TA$)=3 THEN 1470
1460 RETURN
1470 K=K*100:GOTO 1460
1480 K=K*10
1490 GOTO 1460
1500 REM another player
1510 PRINT:PRINT:RESTORE
1520 D1=0:R$="NEXT":GOTO 460
1530 REM high point
1535 IF P=1 THEN TP=TP(1):GOTO 1720
1540 MM=0:VV=1:N=N-1
1550 IF TP(VV)>=TP(VV+1) THEN 1590
1560 TP=TP(VV):TP(VV)=TP(VV+1)
1570 TP(VV+1)=TP:N$=N$(VV)
1580 N$(VV)=N$(VV+1):N$(VV+1)=N$:MM=1
1590 IF VV+1>=N THEN 1600       'final
1595 VV=VV+1:GOTO 1550
1600 REM mm=1--not in order
1610 IF MM=1 THEN 1540
1650 REM points compiled(final)
1700 REM print results
1710 PRINT "ALL HAVE PLAYED..."
1720 PRINT "STRIKE A KEY FOR RESULTS ON SCORE"
1730 GOSUB 3000
1740 CLS:I=1
1750 REM single player
1760 IF P=1 THEN 1850
1770 PRINT "THE TOP WINNER: ";N$(I)
1780 PRINT "WITH A FINAL POINT SCORE OF: ";
1790 PRINT USING G$;TP(I)
1800 PRINT "OTHER PLAYERS AND SCORES:"
1810 FOR X=I+1 TO P
1820 PRINT N$(X);" HAD: ";USING G$;TP(X)
1830 NEXT
```

```
1840 GOTO 1880
1850 PRINT Q$;" YOUR TOTAL (FINAL) POINTS"
1860 PRINT "FOR GENERAL STORE: ";
1870 PRINT USING G$;TP
1880 PRINT
1890 PRINT "END OF PROGRAM RUN."
1900 END
2500 REM inflation/deflation
2510 FOR I=1 TO 30
2520 DE=INT(RND*10+1):DF=CSNG(DE/10)
2530 DG=INT(RND*2)
2540 IF DG=2 THEN 2590
2550 REM inflation
2560 PR(I)=PR(I)+DF
2570 NEXT
2580 GOTO 570
2590 REM deflation
2600 PR(I)=PR(I)-DF
2610 IF PR(I)<=0 THEN 2560
2620 GOTO 2570
2800 REM random player
2810 IF N>P THEN EE=1:RETURN
2820 Q=INT(RND*P+1)
2830 IF N(Q)=0 THEN 2820
2840 Y=Q:N(Q)=0
2850 N=N+1:RETURN
2900 REM extra message
2910 PRINT
2920 PRINT "ADD ALL ITEMS TOGETHER"
2930 PRINT "BEFORE INPUTTING AMOUNT."
2940 IF D1>=10 THEN 2970
2950 FOR TL=1 TO 2800:NEXT
2960 CLS:RETURN
2970 PRINT "START WITH PREVIOUS TOTAL."
2980 GOTO 2950
3000 REM INKEY$
3010 X$=INKEY$
3020 IF X$=""THEN 3010
3030 RETURN
4000 REM data/price--item combined
4010 DATA MILK, BREAD, MEAT
4020 DATA MEAT, MEAT, FISH
4030 DATA CANDY, CANDY, CANDY
4040 DATA VEGETABLES,VEGETABLES
4050 DATA POULTRY,POULTRY
4060 DATA LETTUCE,ONIONS
4070 DATA SPINACH,CIGARETTES
4080 DATA CIGARS,BACON
```

```
4090 DATA TABLETS(PAPER),PENS
4100 DATA BATTERIES,BATTERIES
4110 DATA FURNITURE POLISH
4120 DATA CARPET CLEANER
4130 DATA HAND SOAP,HAND SOAP
4140 DATA CRACKERS, MEATS
4150 DATA SODAS(CASE)
4160 REM prices of items
4170 DATA 2.30,0.89,1.98,3.35,5.50,3.65
4180 DATA 0.35,2.05,0.54,0.39,0.45,3.44
4190 DATA 1.99,0.59,1.01,0.79,7.05,5.74
4200 DATA 4.66,1.23,2.44,2.66,0.98,1.97
4210 DATA 2.41,0.99,0.99,1.32,8.99,4.32
```

```
 10 REM program title: rain of terror
 20 CLEAR 200
 22 V$=RIGHT$(TIME$,2): 'use PC clock to seed
    random number generator
 24 V=VAL(V$)
 26 RANDOMIZE V
 30 PRINT:WIDTH 40:KEY OFF:COLOR 14,3,7  :CLS
 35 DEF SEG=&HB800
 40 PRINT TAB(10) "*** RAIN OF TERROR ***":GOSUB 810
 50 PRINT:X$=STRING$(40,32)
 60 INPUT "INSTRUCTIONS (Y/N)";A$
 70 IF A$="N" OR A$="NO" THEN 190
 80 PRINT "YOU WILL BE BOMBARDED FROM THE"
 90 PRINT "SKY BY FALLING BOMBS.  ALL YOU"
100 PRINT "NEED TO DO TO STAY ALIVE IS TO"
110 PRINT "USE THE <LEFT> AND <RIGHT ARROW>"
120 PRINT "KEYS TO KEEP OUT OF THE BOMBS' POINTS"
130 PRINT "OF CONTACT.  IT SOUNDS VERY SIMPLE"
140 PRINT "BUT YOU MUST BE FAST!"
160 PRINT
170 PRINT "STRIKE A KEY TO BEGIN"
180 AA$=INKEY$:IF AA$=""THEN 180
190 CLS
200 REM border/messages
210 LOCATE 1,1:PRINT STRING$(40,219)
220 I$=CHR$(220):GOSUB 510
230 R=0:REM points
240 P=160:PP=P+84:U=P+1440:UV=U:UW=U+78:RR=0
250 POKE P,43:POKE U,32:P=P+4
260 U=U+4:POKE(U-4),32:POKE U,220
270 IF P>=PP THEN POKE U,32:GOTO 240
280 SOUND 440,1:PX=INT(RND*79+160)
285 LOCATE 23,1:PRINT P,PX
300 IF ABS(P-PX)<=8 THEN 320
310 GOTO 250
320 REM a drop
330 PL=P
335 IF PEEK(P)=220 THEN 550
340 POKE P,42:P=P+80:FOR I=1 TO 20:NEXT
350 POKE(P-80),32
360 IF P>PL+1440 THEN 410
370 FOR H=1 TO 10:AA$=INKEY$:IF LEN(AA$)=2 THEN
    A$=MID$(AA$,2,1):K=ASC(A$):GOTO
375 ELSE NEXT
375 IF K=75 THEN K=0:GOTO 420
```

```
380 IF K=77 THEN K=0:GOTO 470
390 R=R+25:RR=RR+25:LOCATE 22,18:PRINT RR:LOCATE
    23,16:PRINT R;
400 IF P=U THEN 550 ELSE GOTO 340
410 LOCATE 21,1:PRINT X$:P=PL:RR=0:GOTO 250
420 POKE U,32:POKE UX,32
430 U=U-4
440 IF U<=UV THEN 480
450 POKE U,220
460 UX=U:U=PL+1440:GOTO 340
470 POKE U,32:POKE UX,32
480 U=U+4
490 IF U>=UW THEN U=UW
500 GOTO 450
510 REM messages
520 LOCATE 22,1:PRINT "POINTS BEFORE HIT ";
530 LOCATE 23,1:PRINT "TOTAL POINTS ";
540 RETURN
550 REM bombed
560 R=R-RR:LOCATE 23,16:PRINT R:LOCATE 21,1:PRINT X$
570 IF U<=1604 THEN U=U+6
580 IF U>=1676 THEN U=U-8
590 U1=U-80:U2=U1-2:U3=U1+2:U4=1
600 POKE U,42
605 SOUND 1000,3
610 POKE U1,42:POKE U2,42
615 SOUND 1040,3
620 POKE U3,42
625 SOUND 1080,3
630 FOR T=1 TO 200:NEXT:U4=U4+2
640 POKE U,32:FOR V=0 TO 18 STEP 2:POKE(U2+V),32
    :NEXT
650 IF ABS(U-U1)>=328 THEN 670
660 U1=U1-80:U2=U1-U4:U3=U1+U4:GOTO 600
670 REM gone
680 FOR T=1 TO 1000:NEXT
690 CLS
700 PRINT "YOU HAVE BEEN DESTROYED BY ONE"
710 PRINT "OF THE FALLING BOMBS.  YOUR"
720 PRINT "TOTAL POINTS ACCRUED BEFORE"
730 PRINT "BEING DESTROYED WERE:";R
740 PRINT:FOR T=1 TO 1000:NEXT
750 A$="END OF RAIN OF TERROR."
760 FOR I=1 TO LEN(A$)
770 PRINT MID$(A$,I,1);:SOUND 220,1:SOUND 32767,1
790 NEXT
800 END
```

```
810 PLAY "MB O2 L12 FCF L4 A- F L12 A- F A-  L4
    O3 C O2 A- L12 O3 C O2 A- O3 C L 4 E-O2 E- L12
    A- E- A- O3L2 C"
820 RETURN
```

Appendix B
Program Listings for the Apple II

The following listings have been created especially for use on the Apple II, Apple *II*e, and Apple II plus.

```
 10  REM  PROGRAM TITLE:GENERAL STORE
 20  HOME : HTAB 10
 30  PRINT "GENERAL STORE":R$ = "NEXT"
 40  DIM N$(15),I$(31),I(31)
 50  DIM L(31),PR(31),TP(15): REM
     INSTRUCTIONS
 60  PRINT "YOUR INSTRUCTIONS ARE SIMPLE
     AND"
 70  PRINT "STRAIGHT-FORWARD: I WILL GIVE"
 80  PRINT "YOU A LIST OF DIFFERENT ITEMS"
 90  PRINT "AND THEIR PRICES.  YOU IN TURN"
100  PRINT "WILL ENTER THE TOTAL PRICE OF
     ALL THE"
110  PRINT "ITEMS. THAT IS, IF I SAY $2.50,"
120  PRINT "MILK, AND THEN SAY $0.80 BREAD"
130  PRINT "YOU MUST THEN ENTER THE TOTAL"
140  PRINT "OF $2.50 + $0.80 ($3.30). ALL"
150  PRINT "OF YOUR ENTRIES SHOULD BE
     ENTERED"
160  PRINT "WITHOUT DOLLAR SIGNS--JUST PUT"
170  PRINT "IN THE DECIMAL IF THE NUMBER
     HAS A"
180  PRINT "CENT AMOUNT.": PRINT : PRINT
     "PRESS RETURN TO CONTINUE:";
190  GOSUB 3000: HOME :G$ = "$"
210  PRINT "AFTER TEN ITEMS WE WILL"
220  PRINT "TAKE A SHORT BREAK, THEN I WILL"
230  PRINT "GIVE YOU TEN MORE. WE WILL STOP"
240  PRINT "ONLY AFTER THIRTY ITEMS HAVE
     BEEN"
250  PRINT "TOTALLED.  YOUR POINTS WILL BE"
260  PRINT "DETERMINED BY THE ACCURACY OF
     YOUR"
270  PRINT "TOTAL.  IF CORRECT, THE TOTAL
     WILL BE"
280  PRINT "YOUR POINTS.  IF INCORRECT,ALL"
290  PRINT "MISSED WILL BE SUBTRACTED FROM
     YOUR TO-"
300  PRINT "TAL.  YOU WILL HAVE A TIME
     LIMIT IN"
310  PRINT "WHICH TO ENTER THE TOTAL,SO LAY"
320  PRINT "DOWN THE PENCIL AND THE
     CALCULATOR!"
340  REM  BEGIN
350  PRINT "PRESS ANY KEY TO BEGIN:"
360  GOSUB 3000: HOME
```

```
370  INPUT "HOW MANY PLAYERS (1-10)?";P
380  IF P < 1 OR P > 10 THEN 370
390  PRINT "ENTER ";P;" PLAYERS' FIRST NAMES:"
400  REM  NAMES
410 AP = 1310: FOR I = 1 TO P
420  INPUT N$(I):N(I) = I: NEXT :N = 1:A1
     = 3:A2 = 30
425  AQ = AP + 128: IF AQ >  = 1959 THEN
     AQ = 1094
427  IF P = 1 THEN Y = 1:N = 2: GOTO 470
430  PRINT "I WILL SELECT EACH PLAYER IN"
440  PRINT "A RANDOM FASHION.  THERE WILL"
450  PRINT "BE NO SET ORDER OF PLAYERS."
460  GOSUB 2800: PRINT : REM  PLAYER
470  IF EE = 1 THEN 1530: REM  FINAL
480 Q$ = N$(Y):TA = 0:JI = 0
490  PRINT Q$;" WILL NOW PLAY:"
500  PRINT "PRESS THE RETURN KEY TO BEGIN";
510  INPUT X$: PRINT
520  HOME : PRINT "HOW FAST IS YOUR
     ADDITION,"
530  PRINT Q$;"???"
540  REM  READ ALL ITEMS/PRICES
550  FOR I = 1 TO 30: READ I$(I):L(I) = I:
     NEXT
560  FOR I = 1 TO 30: READ PR(I): NEXT :
     GOTO 2500
570 D = I - 1:DD = (D / 3):U = 1:D$ = "$":
     F$ = "$"
580  REM  SELECT ITEMS RANDOMLY
590  IF U > DD THEN  PRINT : PRINT : GOTO
     1100
600 X =  INT ( RND (1) * 30 + 1)
610  IF L(X) = 0 THEN 600: REM  USED
620 T = X:PR = PR(T):T$ = I$(T)
625  IF U = 1 THEN YU = PR
630 TA = TA + PR: REM  TOTAL AMOUNT/CURRENT
     ITEM
640 TA$ =  STR$ (TA): REM  TOTAL TO STRING
645  IF U = 1 THEN  GOSUB 2900
650  IF U = 1 THEN  HOME
655  IF U > 1 THEN 670
660  HTAB 1: PRINT "PRICES";: HTAB 12:
     PRINT "ITEMS";
655  IF U > 1 THEN 670
660  HTAB 1: PRINT "PRICES";: HTAB 12:
     PRINT "ITEMS";
```

```
  665  HTAB 30: PRINT "PRICES
  670  HTAB 1: PRINT D$;PR;
  680  HTAB 12: PRINT T$
  690  IF U (  = 1 THEN 710
  700 RR = 1: REM  CONTINUE PRINT/ENTRIES
  710 L(X) = 0:U = U + 1:KK = 0:NE = 1
  720  IF RR (  ) 1 THEN 580
  725  VTAB A1: HTAB A2: PRINT F$
  730 XS =  PEEK ( - 16384) - 128: IF KK (  =
      5 AND XS ) 0 THEN  POKE ( - 16368),0 :
      GOTO 1040 !
  735  IF XS = 46 THEN  POKE ( - 16368),0:
      GOTO 880
  740  IF XS (  = 0 THEN 760
  750  POKE ( - 16368),0: GOTO 800
  760 KK = KK + 1: REM  TIME EXPIRED?
  770  IF KK )  = 250 THEN W$ = "": GOTO 1000
  780  POKE AP,170: GOTO 730
  800  REM  ENTRY
  810  POKE AP,XS + 128:TY(NE) = XS
  820 AP = AP + 1: REM  LOCATION
  830 NE = NE + 1: REM  NUMBER ENTRY
  840  IF NE =  LEN (TA$) THEN 860
  850  GOTO 730
  860  GOTO 910: REM  ALL ENTRIES IN
  880  REM  DECIMAL
  890  POKE AP,XS + 128
  900 TY(NE) = 1000: GOTO 820
  910  REM  COMPILE ENTRIES
  920 V = NE - 1: IF V = 9 THEN V = 10
  930  FOR I = 1 TO V
  940  IF TY(I) = 1000 THEN 970
  950 W$ = W$ +  STR$ (TY(I))
  960 TY(I) = 0: NEXT : GOTO 990
  970 W$ = W$ + "."
  980  GOTO 960
  990 W =  VAL (W$):W$ =  STR$ (W)
 1000  IF W$ (  ) TA$ THEN PI = PI + PR:JI =
      JI + 1
 1005  IF U = 3 AND JI (  ) 0 THEN PI = PI +
      PR:JI = JI + 1
 1008  IF  PEEK (AP) = 170 THEN  POKE AP,160
 1010  POKE AP + 2,160:W$ = "":A1 = A1 + 1
 1020 AP = AQ:AQ = AQ + 128: IF AQ )  = 1959
      THEN AQ = 1094
 1030  GOTO 590: REM  ANOTHER ITEM
 1040  REM  SLOW DOWN A LITTLE
 1050  POKE AP,160
```

```
1060  GOTO 730
1100  REM  POINTS/PLAYER ADVANCE
1110  IF PI <  > 0 THEN 1200
1120  FOR I = 1 TO  LEN (TA$)
1130  IF  MID$ (TA$,I,1) = "." THEN 1150
1140 K$ = K$ +  MID$ (TA$,I,1)
1150  NEXT
1160 K =  VAL (K$)
1170  GOSUB 1430
1180 TP(Y) = TP(Y) + K
1190  GOTO 1320
1200  REM  MISSED ITEMS
1210 TA$ =  STR$ (PI): FOR I = 1 TO LEN(TA$)
1220  IF  MID$ (TA$,I,1) = "." THEN 1240
1230 K$ = K$ +  MID$ (TA$,I,1)
1240  NEXT :K =  VAL (K$): GOSUB 1430
1250 TP(Y) = TP(Y) - K
1260  PRINT "YOU HAVE MISSED TOTALS ON ";JI
1270  PRINT "ITEMS, ";Q$;". THESE ITEMS CAME"
1280  PRINT "TO A TOTAL OF $";TA$;". YOU ARE"
1290  PRINT "MINUS THAT AMOUNT.  YOUR SCORE"
1300  PRINT "IS NOW ";G$;TP(Y)
1310  GOTO 1350
1320  PRINT "YOU HAVE SUCCESSFULLY ENTERED"
1330  PRINT "ALL TOTALS, ";Q$;". YOUR POINTS"
1340  PRINT "ARE NOW: ";G$;TP(Y)
1350 K = 0:JI = 0:PI = 0:TA$ = "":K$ = ""
1360 U = 1:D1 = D1 + 10: IF D1 = 20 THEN R$
     = "LAST"
1370 AP = 1310:AQ = AP + 128:A1 = 3:A2 =
     30:RR = 0: PRINT
1380  IF D1 )  = 30 THEN 1500
1390  PRINT "TAKE A SHORT BREAK, THEN WE'LL"
1400  PRINT "ATTEMPT THE ";R$;" SET OF ITEMS."
1410  FOR I = 1 TO 5000: NEXT : PRINT
1420  GOTO 580
1430  REM  AMOUNT TIMES 10
1440  IF  MID$ (TA$, LEN (TA$) - 1,1) = "."
      THEN 1480
1450  IF  LEN (TA$) = 2 OR  LEN (TA$) = 3
      THEN 1470
1460  RETURN
1470 K = K * 100: GOTO 1460
1480 K = K * 10
1490  GOTO 1460
1500  REM  ANOTHER PLAYER
1510  PRINT : PRINT : RESTORE
```

```
1520 D1 = 0:R$ = "NEXT": GOTO 460
1530  REM  HIGH POINT
1535  IF P = 1 THEN TP = TP(1): GOTO 1720
1540 MM = 0:VV = 1:N = N - 1
1550  IF TP(VV) > = TP(VV + 1) THEN 1590
1560 TP = TP(VV):TP(VV) = TP(VV + 1)
1570 TP(VV + 1) = TP:N$ = N$(VV)
1580 N$(VV) = N$(VV + 1):N$(VV + 1) = N$:MM = 1
1590  IF VV + 1 > = N THEN 1600: REM  FINAL
1595 VV = VV + 1: GOTO 1550
1600  REM  NOT IN ORDER
1610  IF MM = 1 THEN 1540
1650  REM  POINTS COMPILED (FINAL)
1700  REM  PRINT RESULTS
1710  PRINT "ALL HAVE PLAYED..."
1720  PRINT "PRESS RETURN FOR RESULTS ON
      SCORE"
1730  GOSUB 3000
1740  HOME :I = 1
1750  REM  SINGLE PLAYER
1760  IF P = 1 THEN 1850
1770  PRINT "THE TOP WINNER: ";N$(I)
1780  PRINT "WITH A FINAL POINT SCORE OF: ";
1790  PRINT G$;TP(I)
1800  PRINT "OTHER PLAYERS AND SCORES:"
1810  FOR X = I + 1 TO P
1820  PRINT N$(X);" HAD: ";G$;TP(X)
1830  NEXT
1840  GOTO 1880
1850  PRINT Q$;" YOUR TOTAL (FINAL) POINTS"
1860  PRINT "FOR GENERAL STORE ARE:";
1870  PRINT G$;TP
1880  PRINT
1890  PRINT "END OF PROGRAM RUN"
1900  END
2500  REM  INFLATION/DEFLATION
2510  FOR I = 1 TO 30
2520 DE =  INT ( RND (1) * 10 + 1):DF =
      CSNG(DE / 10)
2530 DG =  INT ( RND (1) * 2 + 1)
2540  IF DG = 2 THEN 2590
2550  REM  INFLATION
2560 PR(I) = PR(I) + DF
2570  NEXT
2580  GOTO 570
2580  GOTO 570
2590  REM  DEFLATION
2600 PR(I) = PR(I) - DF
```

```
2610  IF PR(I) (  = 0 THEN 2560
2620  GOTO 2570
2800  REM  RANDOM PLAYER
2810  IF N ) P THEN EE = 1: RETURN
2820 Q =  INT ( RND (1) * P + 1)
2830  IF N(Q) = 0 THEN 2820
2840 Y = Q:N(Q) = 0
2850 N = N + 1: RETURN
2900  REM  EXTRA MESSAGE
2910  PRINT
2920  PRINT "ADD ALL ITEMS TOGETHER"
2930  PRINT "BEFORE INPUTTING AMOUNT."
2940  IF D1 )  = 10 THEN 2970
2950  FOR TL = 1 TO 2800: NEXT
2960  HOME : RETURN
2970  PRINT "START WITH PREVIOUS TOTAL."
2980  GOTO 2950
3000  REM  INPUT
3010 XS =  PEEK ( - 16384) - 128
3020  IF XS (  = 0 THEN 3010
3030  POKE ( - 16368),0: RETURN
4000  REM  DATA/PRICE--ITEM COMBINED
4010  DATA  MILK,BREAD,MEAT
4020  DATA  MEAT,MEAT,FISH
4030  DATA  CANDY,CANDY,CANDY
4040  DATA  VEGETABLES,VEGETABLES
4050  DATA  POULTRY,POULTRY
4060  DATA  LETTUCE,ONIONS
4070  DATA  SPINACH,CIGARETTES
4080  DATA  CIGARS,BACON
4090  DATA  TABLETS(PAPER),PENS
4100  DATA  BATTERIES,BATTERIES
4110  DATA  FURNITURE POLISH
4120  DATA  CARPET CLEANER
4130  DATA  HAND SOAP, HAND SOAP
4140  DATA  CRACKERS, MEATS
4150  DATA  SODAS (CASE)
4160  REM  PRICES OF ITEMS
4170  DATA   2.35,1.09,1.98,3.35,5.55,3.65
4180  DATA   1.35,2.15,1.04,1.39,1.45,3.44
4190  DATA   1.99,1.09,1.11,1.29,7.15,5.74
4200  DATA   4.66,1.23,2.44,2.66,.98,1.97
4210  DATA  2.41,1.99,1.99,1.32,8.99,4.32

]
```

```
10  PRINT  CHR$ (4);"BLOADSOUND"
20 F = 200
105  HOME
125  HTAB 15: PRINT "HOLD TIME"
130  PRINT : INPUT "INSTRUCTIONS? ";Z$
135  IF Z$ = "N" OR Z$ = "NO" THEN  HOME :
     GOTO 275
140  PRINT : HOME
145  PRINT "OLD TIME...A MATTER OF HIGH OR"
150  PRINT "LOW POINT TOTALS. THE COMPUTER'S"
155  PRINT "CRAFT WILL BLIP ALL AROUND YOUR"
160  PRINT "VIDEO.  DOWN THE LEFT AND RIGHT"
165  PRINT "SIDES WILL BE ASTERISKS.  THESE"
170  PRINT "ARE YOUR LASERS. AT THE INSTANT"
175  PRINT "YOU THINK THE CRAFT IS IN LINE"
180  PRINT "WITH THESE, HOLD THE SPACE BAR.
     YOUR"
185  PRINT "LASER BLASTS WILL START MOVING"
190  PRINT "TOWARD THE CENTER. THE COMPUTER"
195  PRINT "CRAFT WILL ALSO CONTINUE TO
     MOVE."
200  PRINT : INPUT "PRESS RETURN TO
     CONTINUE:";Z$
205  PRINT "YOU CAN 'HOLD' THE COMPUTER'S"
210  PRINT "CRAFT FROM MOVING FOR A SHORT
     TIME,"
215  PRINT "ROUGHLY 20 SECONDS, BY HOLDING THE"
220  PRINT "(P) AND (REPT) KEYS.  POINTS
     WILL RANGE"
225  PRINT "1500 TO 400, DEPENDING ON
     WHERE ON"
230  PRINT "THE SCREEN (TOP OR BOTTOM) YOU
     DESTROY"
235  PRINT "THE CRAFT.  TIME LIMIT FOR THE"
240  PRINT "ENTIRE RUN WILL BE JUST ABOUT"
245  PRINT "FIVE MINUTES. THIS WILL BE NOTED"
250  PRINT "AT THE LOWER RIGHT OF YOUR
     SCREEN."
255  PRINT "COUNT STARTS AT 30."
260  PRINT
275  PRINT : INPUT "PRESS RETURN TO BEGIN "
     ;X$: HOME
295 C$ = "X"
305 Q$ = "*":MR = 8:MC = 16:M1 = 1:M2 = 1
310 G = 0:PI = 16: FOR X = 1 TO 40: VTAB 1:
     HTAB X: PRINT Q$;: VTAB 16: HTAB X:
```

```
     PRINT Q$;: NEXT
315  VTAB 2: HTAB 36: PRINT "20";: FOR XR =
     2 TO 15:XC = 1
320  VTAB XR: HTAB XC: PRINT Q$;: HTAB XC +
     39: PRINT Q$;
323  POKE 769,93: POKE 770,5: POKE 768,1:
     CALL 768
325  GOTO 500
330  IF M1 (  ) 1 AND M2(  )1THEN  GOSUB 355
333  NEXT
335  FOR X = 1 TO 40: VTAB 1: HTAB X: PRINT
     CHR$ (32);: VTAB 16: HTAB X: PRINT CHR$
     (32);
337  NEXT
340  FOR XR = 2 TO 15:XC = 1
345  VTAB XR: HTAB XC: PRINT  CHR$ (32);:
     HTAB 40: PRINT  CHR$ (32);
350  GOSUB 355: NEXT : GOSUB 355: GOTO 310
355  VTAB 16: HTAB 35: PRINT  ABS ( INT (TT
     / 60) - 30);
356  VTAB M1: HTAB M2: PRINT "   ";
360  IF MR (  = 2 OR MR )  = 15 THEN MR = 7
361  VTAB MR: HTAB MC: PRINT C$;:TT = TT + 1
362  IF TT )  = 1800 THEN 830
365 M1 = MR:M2 = MC:W =  PEEK ( - 16384) -128
366  IF W = 80 AND G (  = 20 THEN G = G + .
     4: GOSUB 770: RETURN
370 XX =  INT ( RND (1) * 4 + 1): REM
     CRAFT MOVE
375  ON XX GOTO 380,390,400,410
380  REM  LEFT MOVE
385 MC = MC -  INT (RND (1) * 5 + 1):GOTO 420
390  REM  RIGHT MOVE
395 MC = MC +  INT ( RND (1)*5+1):GOTO 420
400  REM  UP MOVE
405 MR = MR - 1: GOTO 420
410  REM  DOWN MOVE
415 MR = MR + 1: IF  PEEK (1153) = 32 THEN
     GOSUB 700: REM  LOCATION OF MESSAGE
     FROM  LINES 745-762
416  IF  PEEK (1186) = 32 THEN  GOSUB 700:
     REM   LOCATION OF MESSAGE FROM LINE 315
417  IF  PEEK (1835) = 32 THEN  GOSUB 700:
     REM   LOCATION OF MESSAGE FROM LINE 705
418  IF  PEEK (1859) = 32 THEN  GOSUB 700:
     REM   LOCATION OF MESSAGE FROM LINE 715
420  IF MC (  = 3 THEN MC = 38:MR = MR - 1
```

```
422  IF MR (  = 2 AND MC (  = 3 THEN 410
425  IF MC )  = 35 THEN MC = 3:MR = MR + 1
427  IF MR )  = 15 AND MC )  = 35 THEN 400
430  RETURN : REM  SOUND ROUTINE LOCATION
500  REM  FIRE
505  IF XR (  = 14 THEN W =  PEEK ( - 16384)
     - 128
510 PI = PI - 1: IF W (  = 0 THEN 330
515  IF W = 32 THEN  POKE ( - 16368),0:
     GOTO 525
520  GOTO 330
525  REM  LASER ADVANCE
530  IF XR )  = 13 THEN 330
532  IF XR (  = 2 THEN 330
535 HR = XR:HC = XC: IF HR )  = 2 AND HR (
     = 8 THEN H1 = HR * 128 + 896
536  IF HR )  = 9 AND HR (  = 16 THEN H1 =
     (HR - 8) * 128 + 936
537 H2 = H1 + 39:UN = UN + 1: REM  SOUND
     ROUTINE
538  IF RP = 1 THEN  VTAB 2: HTAB 20: PRINT
     ABS (R - UN);
540  POKE H1,160: POKE H2,160
545  IF  PEEK (H1 + 1) (  ) 160 OR  PEEK
     (H2 - 1) (  ) 160 THEN 570
550 H1 =H1 + 1:H2 = H2 -1: REM SOUND ROUTINE
551  POKE 769,112
552  POKE 770,6
553  POKE 768,1
554  CALL 768
555  POKE H1,170: POKE H2,170
560  IF  ABS (H2 - H1) (  = 2 THEN 900
565  GOSUB 355: GOTO 540
570  REM  DIRECT HIT
575  POKE H1,170
580  IF  PEEK (H2 - 1) (  ) 160 THEN 600
585  POKE H2,160:H2 = H2 - 8
590  POKE H2,170
595  FOR O = 1 TO 2:NEXT:REM  SOUND ROUTINE
596  GOTO 580: REM  TACK ONTO LINE 595
600  REM  DELETE
605 Z = 1: POKE H1,160: POKE H2,160
610  IF MR )  = 2 AND MR (  = 8 THEN N = MR
     * 128 + 896 + MC - 1: REM  SOUND ROUTINE
612  IF MR )  = 9 AND MR (  = 16 THEN N =
     (MR - 8) * 128 + 936 + MC - 1
615  POKE N,160: POKE N + 1,160
620  POKE N + 2,160
```

```
625 J =  INT ( RND (1) * 5 + 1)
630  IF N + J - 128 )  = 1190 THEN 640
635  GOTO 660
640  POKE N + J - 128,170: POKE N + 128,170
645  FOR I = 1 TO J * 10: NEXT
650  POKE N + J - 128,160: POKE N + 128,160
655  GOTO 675
660  POKE N + J + 128,170: POKE N + 128,170
665  FOR I = 1 TO J * 10: NEXT
670  POKE N + J + 128,160: POKE N + 128,160
675  IF Z (  = 7 THEN 685
680  IF RP = 0 THEN  GOTO 690
682 PI = PI + 1: GOTO 695
685 Z = Z + 1: GOSUB 1000
687  GOTO 625
690 PI = PI + 1:X$ = X$ + "+"
695 PP = PP + PI * 100: GOSUB 700: GOTO 900
700 A$ = "TOTAL POINTS:":T$ = "TIME LAPSE: "
705 FR = 15:FC = 3
710  FOR I = 1 TO  LEN (A$)
715  VTAB FR: HTAB FC+25:PRINT MID$(T$, I, 1);
716  VTAB FR: HTAB FC: PRINT  MID$ (A$, I, 1);
720 FC = FC + 1: NEXT
725  PRINT PP;: IF RP = 1 THEN 735
730  IF X$ = "" THEN  RETURN
735  REM  RESERVE POWER
740 R$ = "RESERVE POWER: "
745 FR = 2:FC = 2
750  FOR I = 1 TO  LEN (R$)
755  VTAB FR: HTAB FC: PRINT  MID$ (R$, I, 1);
760 FC = FC + 1: NEXT
762  IF RP = 1 THEN X$ = STR$ ( ABS (R - UN))
765  PRINT X$;
770 TI$ = "HOLD TIME ": REM  HOLD TIME
775  IF  PEEK (1864) (  ) 32 THEN 790
780 FR = 2:FC = 25: FOR I = 1 TO  LEN (TI$)
785  VTAB FR: HTAB FC: PRINT  MID$ (TI$,I,1)
     ;:FC = FC + 1: NEXT
790  VTAB 2: HTAB 36: PRINT  INT ( ABS (G -
     20));
795  POKE ( - 16368),0: RETURN
800  REM  TIME REMAINING
805  IF RP = 1 THEN DK = 1: GOTO 910
810  IF TT )  = 1800 THEN 830
820  GOTO 300
830  FOR TI = 1 TO 1200: NEXT
835  HOME :R =  LEN (X$)
```

```
840  IF R = 0 THEN 960
845  PRINT "VERY WELL DONE..."
850  PRINT "YOU HAVE ";R;" UNITS OF"
855  PRINT "RESERVE POWER...."
860  PRINT "PRESS RETURN TO USE";
865  INPUT X$:TT = 0:RP = 1:UN = 0
870  HOME
875  GOTO 300
900  POKE H1,160: POKE H2,160: REM  DELEGE
     RESERVE UNITS
905  IF RP = 0 THEN 330
910  IF UN >  = R THEN 925
915  IF DK = 0 THEN 330
920 DK = 0: GOTO 810
925  REM  FINISH
930  FOR TI = 1 TO 1200: NEXT
935  HOME
940  PRINT "RESERVE POWER EXHAUSTED...."
945  PRINT "TOTAL POINTS ACCUMULATED: ";PP
950  PRINT
955  GOTO 980
960  PRINT "YOU HAVE DESTROYED NOTHING"
965  PRINT "YOU HAVE NO RESERVE POWER."
970  PRINT
975  GOTO 945
980  PRINT
985  PRINT "END OF RUN."
990  END
1000 F = F + 7
1005  POKE 769,F
1010  POKE 770,7
1020  POKE 768,2
1030  CALL 771
1040  RETURN

  ]
```

```
1  REM  PROGRAM TITLE BE PREPARED
10  PRINT  CHR$ (4);"BLOADSOUND"
20  HOME : HTAB 10: PRINT "))BE PREPARED ((" 
30  PRINT "BE SURE YOU HAVE READ THE
    INSTRUCTIONS
40  PRINT "BEFORE YOU START.DO YOU NEED TO "
50  INPUT "SEE THEM NOW (Y/N)? ";AA$
60  IF AA$ = "N" THEN 95
70  HOME : GOSUB 2000
90 K = 127
95  GR
100  FOR I = 1282 TO 1287
110  FOR J = I TO I + 30 STEP 10
120  POKE J,K
130  NEXT : NEXT
200  FOR I = 1576 TO 1578
210  POKE I,K: POKE I + 37,K
220  NEXT
300  FOR I = 1583 TO 1588
310  FOR J = I TO I + 20 STEP 10
320  POKE J,K
330  NEXT : NEXT
400  FOR I = 1410 TO 1922 STEP 128
410  FOR J = I TO I + 35 STEP 5
420  POKE J,K
430  NEXT : NEXT
500  FOR I = 1066 TO 1450 STEP 128
510  FOR J = I TO I + 35 STEP 5
520  POKE J,K
530  NEXT : NEXT
1000  POKE  - 16368,0: IF C ) 9 THEN C = 1
1005  IF C ) 9 THEN C = 1
1006  IF VD = 0 AND VW = 0 THEN F = 10000
1007  IF VD + VW = 20 THEN 9400
1010 D =  INT ( RND (1) * C + 1)
1015  IF F (  = 1 THEN F = 0: GOTO 9300
1017  GOSUB 9500
1030 S = 1832
1040  IF S = 1871 THEN DI =  - 1: GOTO 1090
1050  IF ( PEEK ( - 16384) - 176) = D THEN
      7000
1060 T = S:S = S + 1
1070  POKE T,0: POKE S,153:F = F - 10
1080  GOTO 1040
1090 T = S:S = S - 1
```

```
1093  IF ( PEEK ( - 16384) - 176) = D THEN
      POKE T,0: GOTO 7000
1095  IF S = 1832 THEN DI = 1: IF C = 9 THEN
      1000
1096  IF DI =  - 1 THEN 1100
1097  IF C <  > 9 AND S = 1832 THEN C = C +
      1: POKE  - 16368,0: GOTO 1005
1100  POKE T,0: POKE S,153:F = F - 10
1110  IF DI =  - 1 THEN 1090
2000  PRINT " A LONG TIME FROM NOW, IN A
      GALAXY"
2010  PRINT "LIGHT-YEARS AWAY..."
2020  PRINT : PRINT "YOU HAVE BEEN ASSIGNED
      TO DUTY ABOARD"
2030  PRINT "THE SPACE CARRIER 'XENOPHON'."
2040  PRINT "YOUR ASSIGNMENT: TO SAFELY DOCK
      TWENTY"
2050  PRINT "BATTLE CRUISERS RETURNING FROM
      DEEP"
2060  PRINT "SPACE.": PRINT
2070  INPUT "PRESS RETURN TO CONTINUE";AA$
2080  HOME : PRINT "THE VESSELS HAVE
      LIMITED FUEL REMAINING"
2090  PRINT "SO YOU MUST DOCK THEM AS SOON AS"
2100  PRINT "POSSIBLE.  YOU WILL SEE THE PORTS"
2110  PRINT "ON THE SCREEN AS BLUE AND
      WHITE TUBES."
2120  PRINT "WHEN YOU THINK THE VESSEL IS
      AT THE"
2130  PRINT "OPENING, PRESS A NUMBER KEY
      BETWEEN"
2140  PRINT "ONE AND TWO FOR THE FIRST
      VESSEL."
2150  INPUT "PRESS RETURN TO CONTINUE ";AA$
2160  HOME : PRINT "FOR THE SECOND VESSEL
      YOU MUST CHOOSE"
2170  PRINT "BETWEEN KEYS ONE THROUGH THREE,
      THE"
2180  PRINT "POSSIBLE CHOICES INCREASING
      FOR EACH"
2190  PRINT "VESSEL UP TO NINE, THEN
      STARTING OVER."
2200  PRINT"IF YOU FIRE AT THE WRONG TIME OR"
2210  PRINT "ALLOW THE VESSEL TO RUN OUT
      OF FUEL,"
2220  PRINT "YOU WILL HAVE DESTROYED A
      VALUABLE"
2230  PRINT "VESSEL AND MANY HUMAN LIVES.
      TAKE THIS"
```

```
2240  PRINT "ASSIGNMENT SERIOUSLY.  THAT
      WILL BE ALL"
2250  PRINT : INPUT "PRESS RETURN TO BEGIN"
      ;AA$
3000  RETURN
7000  POKE  - 16368,0:X = S
7005 AA = 0
7010 AA = AA + 1: GOSUB 8000
7020  IF AA < 11 THEN 7040
7030  GOTO 9200
7040  GOTO 7010
8000  IF X > 1066 AND X < 1101 THEN 9000
8005  IF  PEEK (X - 128) <  > 0 THEN 9100
8010 Y = X:X = X - 128
8020  POKE Y,0: POKE X,153
8030  RETURN
9000  FOR II = 1066 TO 1096 STEP 10
9010  IF X > II AND X < II + 5 THEN Y = X:X
      = X + 856
9020  NEXT II: GOTO 8020
9100  GOSUB 9600
9110  HOME : VTAB 23: FLASH : HTAB 12:
      PRINT "VESSEL DESTROYED":VW = VW + 1:
      FOR TT = 1 TO 2000: NEXT :F = 10000:
      NORMAL : POKE X,0
9115  GOSUB 9500
9120 C = C + 1: POKE  - 16368,0: GOTO 1005
9200  GOSUB 9800
9210  HOME : VTAB 23: INVERSE : HTAB 13:
      PRINT "VESSEL DOCKED": FOR TT = 1
      TO 2000: NEXT : NORMAL : POKE X,0
9220 VD = VD + 1:F = 10000
9225  GOSUB 9500
9230 C = C + 1: POKE  - 16368,0: GOTO 1005
9300  GOSUB 9700
9310  NOTRACE : VTAB 23: INVERSE : HTAB 13:
      PRINT "FUEL EXHAUSTED": FOR TT = 1 T
      O 2000: NEXT : NORMAL :VW = VW + 1:F
      = 10000
9311  POKE X,0
9315  GOSUB 9500
9320 C = C + 1: POKE  - 16368,0: GOTO 1005
9400  GOSUB 9500
9410  VTAB 21: PRINT  TAB( 12);"G A M E
      O V E R          "
9415  FOR TT = 1 TO 6000: NEXT
9420  TEXT : HOME : END
```

```
9500  HOME : VTAB 22: PRINT  TAB( 13);"FUEL
      REMAINING: ";F
9510  VTAB 24: PRINT "VESSELS DOCKED: ";VD;"
      VESSELS DESTROYED: ";VW
9520  RETURN
9600  FOR II = 1 TO 4
9610  POKE 769,159: POKE 770,7: POKE 768,3
9620  CALL 768
9630  POKE 769,173: POKE 770,6: POKE 768,3
9640  CALL 768
9650  NEXT
9660  RETURN
9700  POKE 769,8: POKE 770,14
9705  POKE 768,3
9710  CALL 768
9720  POKE 769,13: POKE 770,30
9725  POKE 768,3
9730  CALL 768
9760  POKE 769,26: POKE 770,49: POKE 768,3
9770  CALL 768
9780  RETURN
9800  POKE 769,6: POKE 770,8
9810  POKE 768,16
9820  CALL 771
9830  POKE 769,135: POKE 770,5
9840  POKE 768,16
9850  CALL 768
9860  POKE 769,93: POKE 770,5
9870  POKE 768,17
9880  CALL 768
9890  RETURN
22140  PRINT "ONE AND TWO FOR THE FIRST
      VESSEL.": PRINT
```

```
10  REM  PROGRAM TITLE:RAIN OF TERROR
20  CLEAR
30  HOME : PRINT
40  HTAB 10: PRINT "*** RAIN OF TERROR ***"
50  PRINT
60  INPUT "INSTRUCTIONS (Y/N)";A$
70  IF A$ = "N" THEN 190
80  PRINT "YOU WILL BE BOMBARDED FROM THE "
90  PRINT "SKY BY FALLING BOMBS.  ALL YOU"
100  PRINT "NEED TO DO TO STAY ALIVE IS TO"
110  PRINT "USE THE LEFT AND RIGHT ARROW
     KEYS"
120  PRINT "TO GET OUT OF THE WAY OF THE"
130  PRINT "FALLING BOMBS.  SOUNDS VERY"
140  PRINT "SIMPLE, BUT YOU MUST BE FAST!"
150  PRINT
160  PRINT
170  PRINT "PRESS RETURN TO BEGIN"
180  INPUT AA$
190  HOME
200  REM  BORDER / MESSAGES
210  GR : COLOR= 7: HLIN 0,39 AT 0
220  GOSUB 510
230 R = 0: REM  POINTS
240 PR = 1:PC = 0:PP = PC + 39:UR = PR +
    13:UC = PC:UV = UC:UW = UC + 34:RR = 0
248 RR = 0
250  COLOR= 3: PLOT PC,PR: COLOR= 0: PLOT
    UC,UR:PC = PC + 2
260 UC = UC + 2: PLOT PC - 2,PR: COLOR= 11:
    PLOT UC,UR
270  IF PC )  = PP THEN  COLOR= 0: PLOT UC,
    UR: GOTO 240
280 PX =  INT ( RND (1) * 39)
300  IF  ABS (PC - PX) (  = 8 THEN 320
310  GOTO 250
320  REM  A DROP
330 PL = PR
340  COLOR= 3: PLOT PC,PR:PR = PR + 1
350  COLOR= 0: PLOT PC,PR - 1
360  IF PR ) PL + 13 THEN 410
370 K =  PEEK ( - 16384) - 128: IF K = 8
    THEN 420
380  IF K = 21 THEN 470
390 R = R + 25:RR = RR + 25: VTAB 21: HTAB
    18: PRINT RR;: VTAB 22: HTAB 18: PRINT R;
```

```
400  IF PR = UR AND PC = UC THEN 550ELSE
     GOTO 340
405  GOTO 340
410 PR = PL:RR = 0: GOTO 250
420  COLOR= 0: PLOT UC,UR: PLOT UX,UR
430 UC = UC - 2
440  IF UC (  = UV THEN 480
450  COLOR= 11: PLOT UC,UR
455  POKE ( - 16368),0:UX = UC
460 UR = PL + 13:UC = PC: GOTO 340
470  COLOR= 0: PLOT UC,UR: PLOT UX,UR
480 UC = UC + 2
490  IF UC )  = UW THEN UC = UW
500  GOTO 450
510  REM  MESSAGES
520  VTAB 21: PRINT "POINTS BEFORE HIT";
530  VTAB 22: HTAB 1: PRINT "TOTAL POINTS ";
540  RETURN
550  REM  BOMBED
560 R = R - RR: HTAB 22: VTAB 18: PRINT R;
570  IF UC (  = 2 THEN UC = UC + 3
580  IF UC )  = 37 THEN UC = UC - 3
590 R1 = UR - 1:C1 = UC:R2 = R1:C2 = C1 -
    1:R3 = R1:C3 = C1 + 1:U4 = 1
600  COLOR=  INT ( RND (1) * 6 + 4): PLOT
    UC,UR
610  PLOT C1,R1: PLOT C2,R2
620  PLOT C3,R3
630  FOR T = 1 TO 200: NEXT :U4 = U4 + 1
640  COLOR= 0: PLOT UC,UR: HLIN C2,C3 AT R2
650  IF  ABS (UR - R1) )  = 4 THEN 670
660 R1 = R1 - 1:C2 = C1 - U4:C3 = C1 + U4:
     GOTO 600
670  REM  GONE
680  FOR T = 1 TO 2500: NEXT
690  TEXT : HOME
700  PRINT "YOU HAVE BEEN DESTROYED BY ONE"
710  PRINT "OF THE FALLING BOMBS.  YOUR"
720  PRINT "TOTAL POINTS ACCRUED BEFORE"
730  PRINT "BEING DESTROYED WERE:  ";R
740  PRINT : FOR T = 1 TO 1000: NEXT
750 A$ = "RAIN OF TERROR"
760  FOR I = 1 TO  LEN (A$)
770  VTAB 15: PRINT  MID$ (A$,I,1);
790  NEXT
800  END
```

```
10  REM  PROGRAM TITLE:KNIGHTS
15  PRINT  CHR$ (4);"BLOADSOUND"
20  CLEAR
30  HOME : HTAB 15
40  PRINT ")))KNIGHTS(((("
50  PRINT
60  PRINT "INSTRUCTIONS NEEDED?"
70  INPUT T$
80  IF T$ = "N" OR T$ = "NO" THEN 370
90  PRINT "THERE ARE 10 KNIGHTS IN A CASTLE."
100  PRINT "THESE ARE KNIGHTS WHO HAVE DONE
     WRONG"
110  PRINT "AND WERE IN A DUNGEON WAITING"
120  PRINT "TO HAVE THEIR HEADS REMOVED."
130  PRINT "THEY HAVE ESCAPED FROM THE
     DUNGEON"
140  PRINT "AND CAPTURED THE CASTLE."
150  PRINT "WITH THEM THEY HAVE EIGHT"
160  PRINT "HOSTAGES WHOM THEY HAVE
     CAPTURED"
170  PRINT "THE KING WILL NOT MEET THEIR
     TERMS"
180  PRINT "SO THEY HAVE VOWED TO FIGHT"
190  PRINT "TO THE DEATH..."
200  PRINT : INPUT "PRESS RETURN TO
     CONTINUE: ";AA$
220  PRINT "YOU ARE THE SHARPSHOOTER
     SELECTED"
230  PRINT "BY THE KING TO TERMINATE EACH"
240  PRINT "OF THESE KNIGHTS.  YOU ARE"
250  PRINT "EQUIPPED WITH A HIGH-POWER
     CROSS-"
260  PRINT "BOW.  ALONG THE CASTLE WALLS
     ARE"
270  PRINT "FIVE WINDOWS.  EACH TIME A
     FIGURE"
280  PRINT "APPEARS AT A WINDOW YOU'LL HAVE"
290  PRINT "THE OPTION OF FIRING (BY
     STRIKING"
300  PRINT "THE SPACE BAR) OR NOT.  THE"
310  PRINT "FIGURE APPEARING IN THE WINDOW"
320  PRINT "COULD BE ONE OF THE HOSTAGES."
330  PRINT "IF IT IS A KNIGHT AND YOU
     DON'T FIRE"
340  PRINT "YOU COULD BE HIT..."
```

```
350  PRINT
360  INPUT "PRESS RETURN TO BEGIN, ";T$
370  HOME :K = 10:HD = 8:CR = 10: GR
380  REM  MAKE A CASTLE
390  COLOR= 6: FOR C = 0 TO 39: HLIN 0,39
     AT C: NEXT
400  COLOR= 10: FOR C = 10 TO 26: HLIN 2,38
     AT C: NEXT
410  FOR C = 2 TO 38 STEP 2
420  PLOT C,9: NEXT
510 X = 13:Y = 4: COLOR= 6
520  FOR C = X TO (X + 10)
530  FOR Y = 4 TO 32 STEP 7
540  HLIN Y,(Y + 4) AT C
550  NEXT Y
560  NEXT C
590  REM  GET A WINDOW
600 W =  INT (5 *  RND (1) + 1)
610 WI = 7 * W - 3
620  POKE ( - 16368),0: REM  PAUSE BEFORE
     PLACING
630 PP =  INT (700 *  RND (1) + 251)
640  FOR PU = 1 TO 700
650  IF PU = PP THEN 670
660  NEXT : GOTO 630
670  GOSUB 1630
680  REM  SHARPSHOOTER TO FIRE
690  FOR TI = 1 TO 100
700 WS =  PEEK ( - 16384) - 128
710  IF WS = 32 THEN 730
720  NEXT
730  POKE ( - 16368),0: REM  KNIGHT OR
     HOSTAGE
740 H1 =  INT (5 *  RND (1) + 1):H2 =  INT
    (5 *  RND (1) + 1):H3 =  INT (5 *  RN
    D (1) + 1)
750 H =  ABS (H1 - H2 - H3)
760  GOSUB 1670
762  IF H = W THEN 1240
765  IF HD <  = 0 THEN  THEN 840
770  IF H <  = 0 OR H >  = 5 AND TI <  =
     99 THEN 790
780  GOTO 840
790  REM  A HOSTAGE
800  IF TI >  = 75 THEN 880
810  GOSUB 1800: PRINT "YOU HAVE JUST
     KILLED ONE OF"
```

```
820  PRINT "OF THE HOSTAGES, TURKEY!!!"
830  GOTO 970
840  IF TI >  = 100 THEN 880
850  PRINT "GOOD SHOOTING!  YOU HAVE"
860  PRINT "ELIMINATED ANOTHER KNIGHT!"
865  GOSUB 1900
870  GOTO 940
880  IF H >  = 0 AND H <  = 2 THEN 1020
885  GOSUB 2100
890  PRINT "THINK FASTER!! YOU HAVE BEEN"
900  PRINT "HIT BY ONE OF THE KNIGHTS!!"
910 HK = HK + 1
920  IF HK >  = 6 THEN 1060
930  GOTO 990
940 K = K - 1: PRINT K;" KNIGHTS REMAIN"
950  IF K <  = 0 THEN 1120
960  GOTO 990
970 HD = HD - 1
980  IF HD <  = 0 THEN 1180
990  FOR TX = 1 TO 1500: NEXT : HOME
995  GOTO 1400
1000  REM  CLEAR PRINT AREA
1010  GOTO 590
1020  IF HD <  = 0 THEN 890
1030  PRINT "GOOD THING YOU DIDN'T SHOOT!"
1040  PRINT "THAT WAS A HOSTAGE!"
1045  GOSUB 1900
1050  GOTO 990
1060  REM  HIT ENOUGH
1080  PRINT "YOU HAVE BEEN HIT ";HK;" TIMES."
1090  PRINT "YOUR BODY CANNOT STAND"
1100  PRINT "ANY MORE.  YOU ARE DYING."
1110  GOTO 1530
1120  REM  KNIGHTS KILLED
1130  GOSUB 1900
1140  PRINT "YOU HAVE KILLED ALL TEN"
1150  PRINT "KNIGHTS.  THE KING WILL BE"
1160  PRINT "PROUD OF YOU, MY FRIEND..."
1165  GOSUB 1900
1170  GOTO 1530
1180  REM  HOSTAGES GONE
1190  GOSUB 1800
1200  PRINT "YOU HAVE KILLED ALL OF THE "
1210  PRINT "HOSTAGES...YOU TURKEY!!"
1220  PRINT "THE KING WILL HAVE YOUR HEAD!!"
1230  GOTO 990
1240  REM  COMPLETE MISS
1250 XX =  INT (37 *  RND (1) + 3)
```

```
1270 YY =  INT (10 *  RND (1) + 13)
1280  IF YY ( , = 17 THEN 1270
1290  COLOR= 0: PLOT XX,YY
1295  GOSUB 2100
1310  PRINT "YOU'LL HAVE TO DO BETTER"
1320  PRINT "THAN THAT, YOU'VE MISSED!!"
1330  GOTO 990
1350  REM  CROSSBOW ROUNDS
1360  VTAB 21: HTAB 1
1370  PRINT "YOU HAVE ";CR;" ARROWS TO FIRE."
1380  RETURN
1400  REM  ENOUGH ARROWS LEFT
1410  IF TI (  = 99 THEN CR = CR - 1
1420  GOSUB 1350: IF CR (  = 0 THEN 1440
1425  FOR TX = 1 TO 2000: NEXT : HOME
1430  GOTO 590
1440  REM  END
1450  FOR TX = 1 TO 2000: NEXT
1460  TEXT : HOME
1465  GOSUB 2100
1470  PRINT "YOU HAVE EXHAUSTED YOUR"
1480  PRINT "SUPPLY OF CROSSBOW ARROWS."
1490  IF K (  ) 0 THEN 1510
1500  GOTO 1530
1510  PRINT "THERE ARE STILL ";K;" KNIGHTS"
1520  PRINT "IN THE CASTLE"
1530  PRINT
1540  IF HD (  = 0 THEN 1560
1550  GOTO 1600
1555  GOSUB 2100
1560  PRINT "YOU ARE A BLUNDERING SHARP-"
1570  PRINT "SHOOTER, YOU IDIOT!!!"
1580  PRINT "YOU HAVE HIT AND KILLED ALL"
1590  PRINT "EIGHT HOSTAGES!"
1600  PRINT
1610  PRINT "END OF RUN"
1620  END
1630  COLOR= 9: FOR Y = (WI + 1) TO (WI + 3)
1640  FOR Z = 21 TO 23: PLOT Y,Z
1650  NEXT Z: NEXT Y
1660  RETURN
1670  COLOR= 6
1680  FOR Y = (WI + 1) TO (WI + 3)
1690  FOR Z = 21 TO 23: PLOT Y,Z
1700  NEXT Z: NEXT Y
1710  RETURN
1800  REM  BEE-BOOP
1810  POKE 769,33
```

```
1820  POKE 770,14
1830  POKE 768,10
1840  CALL 771
1850  POKE 769,13
1860  POKE 770,30
1870  POKE 768,25
1880  CALL 771
1890  RETURN
1900  REM  TA-DA TA-DAA!!
1910  POKE 769,4
1920  POKE 770,23
1930  POKE 768,10
1940  CALL 771
1950  POKE 769,8
1960  POKE 770,14
1970  POKE 768,10
1980  CALL 771
1990  POKE 769,4
2000  POKE 770,23
2010  POKE 768,10
2020  CALL 771
2030  POKE 769,8
2040  POKE 770,14
2050  POKE 768,50
2060  CALL 771
2070  RETURN
2100  POKE 769,200
2110  POKE 770,75
2120  POKE 768,1
2140  RETURN

]
]
```

```
 10  REM  PROGRAM TITLE MIND INVASION
 20  REM  INSTRUCTION ROUTINE
 30  CLEAR : GOSUB 1500:J = 1:TR = 1
 40  GR : REM  OUTER GRAPHICS
 50 X = 1:Y = 38:K = 1:PN = 1:TX = 1939:TL
    = TX:X1 = 1
 60  COLOR= K: PLOT X1,X: PLOT Y,X: PLOT X1
     + 2,X: PLOT Y - 2,X
 70 X = X + 2:Y = Y - 1:J = J + 1:X1 = X1 + 1
 80  IF J <  > 8 THEN  GOTO 60
 90  REM  GOTO CHARACTER PRINT/IF DONE
 95  PLOT X1,X: PLOT Y,X:X1 = X1 - 1:Y =
     Y + 1:X = X + 1
100  IF J = 16 THEN  POKE TX,224:N = 14:
     GOTO 140
110  COLOR= K: PLOT X1,X: PLOT Y,X: PLOT X1
     + 2,X: PLOT Y - 2,X
115  IF PN = 1 THEN A(1) = X1:A(2) = Y:L =
     A(1):R = A(2)
120 X = X + 2:X1 = X1 - 1:Y = Y + 1
130 J = J + 1:PN = 2: GOTO 100
140  VTAB 21: HTAB 23: PRINT "YOUR POINTS= ";
150 PI = 0
160  GOSUB 910
170  PRINT "PLAYER:";N$(PZ)
200  REM  CHARACTER GRAPHICS VARIABLES
210 J = Y:QU = 2:QW = 32:E = 8:F = 10:TC
    = 1:BC = 16:T = 1:E2 = E:F2 = F
220 KY = 1:G = 30: IF TI >  = 3 THEN 880
230  COLOR= 13: FOR I = 0 TO J: PLOT QU +
     I,E: PLOT QW + I,E: PLOT QU + I,F: PLO
     T QW + I,F:E1 = E - 1:F1 = F + 1
240 T = T + 1
250  COLOR= 0: PLOT QU + I,E: PLOT QW + I,
     E: PLOT QU + I,F: PLOT QW + I,F:E1 = E
     - 1:F1 = F + 1
260  IF T >  = 2 THEN J = J - 1
270  COLOR= 13: FOR I = 0 TO J: PLOT QU +
     I,F - 1: PLOT QW + I,F - 1: PLOT QU +
     I,E - 1: PLOT QU + I,E: PLOT QU + I,F
280  PLOT QW + I,E - 1: PLOT QW + I,F + 1:
     PLOT QW + I,E1 - 1: PLOT QW + I,F1 + 1
290  COLOR= 0: PLOT QU + I,F - 1: PLOT QW +
     I,F - 1: PLOT QU + I,E - 1: PLOT QU
     + I,E: PLOT QU + I,F
```

```
300  PLOT QW + I,E - 1: PLOT QW + I,F + 1:
     PLOT QW + I,E1 - 1: PLOT QW + I,F1 +1
310 E = E - 1:F = F + 1:E1 = E2 - 2:F1 = F2
    + 2:F1 = E2 + 2
320  IF T <  = 4 THEN E2 = E2 + 1:F2 = F2
     + 1:T = T + 1
325  IF T >  = 5 THEN E2 = E2 - 1:F2 = F2
     - 1:T = T + 1
330  REM  RESET VARIABLES IF DONE
335  GOTO 700
340  IF T >  = 8 THEN 200
345 KY = KY + 1: GOTO 260
350  REM  FOR CHARACTER CLEAR
400 XS =  PEEK ( - 16384) - 128
405 TI = TI + 1: REM  LIMIT COUNT WITHOUT
    MOVEMENT
410  IF XS <  > 73 AND XS <  > 77 THEN
     POKE ( - 16368),0: GOTO 340
420 TI = 1: REM  PLACEMENT/LIMIT/RESET
430  IF XS = 73 THEN  POKE ( - 16368),0:
     GOTO 460
440  IF XS = 77 THEN  POKE ( - 16368),0:
     GOTO 490
450  GOTO 340
460  POKE TX,0: IF TX = 1083 THEN TX = TX
     + 856
465  IF TX > 1555 AND TX <  = 1939 THEN
     TX = TX - 128
467  IF TX > 1083 AND TX <  = 1339 THEN TX
     = TX - 128
470  IF TX = 1555 THEN TX = TX + 128
480  POKE TX,224:U(KY) = XS: GOTO 340
490  POKE TX,0: IF TX = 1939 THEN TX = TX
     - 856
495  IF TX < 1939 AND TX >  = 1555 THEN TX
     = TX + 128
497  IF TX >  = 1083 AND TX < 1467 THEN
     TX = TX + 128
500  IF TX = 1339 THEN TX = TX - 128
510  POKE TX,224:U(KY) = XS: GOTO 340
600  REM  TOP/BOTTOM CHARACTERS
610  FOR V = 1 TO G: FOR QX = 15 TO 25:
     PLOT QX,TC
620  PLOT QX,BC: NEXT
630  COLOR= 0: FOR QX = 15 TO 25: PLOT QX,TC
640  PLOT QX,BC: NEXT : NEXT
650 TC = TC + 1:BC = BC - 1:G = G - 2
```

```
660  REM  REPLACE USER CHARACTER
670  IF  PEEK (TX) = 0 THEN  POKE TX,238
680  GOTO 310
700  REM  CRUSHER MOVEMENT
705  COLOR= 0: PLOT (A(1) + 2),9: PLOT
     (A(2) - 2),9
710  PLOT L,15: PLOT R,15
720  IF  ABS (L - R) <  = 3 THEN 750
730 L = L + 3:R = R - 3
740  COLOR= K: PLOT L,15: PLOT R,15:GOTO 400
750 L = A(1):R = A(2): GOTO 800
760  POKE L,K: POKE (L + 2),K
770  POKE R,K: POKE (R - 2),K
780  REM  RECYCLE
790  GOTO 400
800  REM  CRUSHED?
810  IF  PEEK (TX) = 0 THEN 840
820  IF TI <  = 2 AND U(KY) <  > U(KY - 1)
     THEN PI = PI +  ABS (TX - TL): GOSUB
910  HTAB 35: VTAB 21: PRINT PI: RETURN
920  GOTO 230
950  REM  PLAYER ADVANCE/END
960 TP(PZ) = PI:PI = 0:PZ = PZ + 1
970  IF PZ > PL THEN  GOSUB 1200:TR = TR +
     1:PZ = 1
980  IF TR >  = 3 THEN  GOSUB 1240: GOTO
     1010
990  FOR LX = 1 TO 14: PRINT : NEXT
1000  GOSUB 2100: HOME :J = 1: GOTO 40
1010  REM  END/HIGH POINT WINNER
1020 T = 0:H = 1: IF H = PL THEN 1120
1030  IF FT(H) >  = FT(H + 1) THEN 1080
1040 FT = FT(H):FT(H) = FT(H + 1)
1050 FT(H + 1) = FT:N$ = N$(H):N$(H) = N$
     (H + 1)
1060 N$(H + 1) = N$
1070 T = 1
1080 H = H + 1
1090  IF H < PL THEN 1030
1100  IF T = 1 THEN 1020: REM  NOT REFINED
1110  GOTO 1300
1120  TEXT
1130  HOME
1140  PRINT "YOUR TOTAL POINTS, AFTER 2"
1150  PRINT "ROUNDS=";TP(H) + WE(H)
1160  GOTO 1390
1200  REM  TALLY FIRST ROUND POINTS
1210  IF TR >  = 2 THEN  RETURN
```

```
1220  FOR I = 1 TO PL
1230 WE(I) = TP(I): NEXT : RETURN
1240  REM  FINAL TALLY
1250  FOR I = 1 TO PL
1260 FT(I) = WE(I) + TP(I)
1270  NEXT
1280  RETURN
1300  REM  TOP WINNER (MORE THAN 1 PLAYER)
1310  HOME
1320 I = 1
1330  PRINT "THE TOP WINNER:";N$(I);" WITH";
1340  PRINT FT(I);" POINTS."
1350  PRINT "FOLLOWED BY:"
1360  FOR I = 2 TO PL: PRINT N$(I);" WITH"
1370  PRINT FT(I);" POINTS."
1380  NEXT
1390  PRINT
1400  PRINT "END OF PROGRAM RUN."
1410  END
1500  REM  INSTRUCTIONS
1510 I$ = "MIND INVASION"
1520 I = 11: HOME
1530  FOR LX = 1 TO  LEN (I$)
1540  HTAB I: PRINT  MID$ (I$,LX,1);
1550  FOR Y = 1 TO 50: NEXT Y
1560 I = I + 1: NEXT LX
1570  FOR Y = 1 TO 1000: NEXT
1580  PRINT
1590  INPUT "PLAYER NUMBER (1-10)";PL
1600  IF PL (  = 0 OR PL )  = 11 THEN 1590
1610  PRINT "PLAY OUTLINE NECESSARY ";
1620  INPUT X$
1630  IF X$ = "N" OR X$ = "NO" THEN 2040
1640  PRINT
1650  PRINT "EACH OF YOU ";PL;" PLAYERS WILL"
1660  PRINT "EXPERIENCE A NEW FORM OF MIND"
1670  PRINT "INVASION.  THE OBJECT OF THE
      GAME:"
1680  PRINT "KEEP PLAYING AS LONG AS
      POSSIBLE"
1690  PRINT "WITHOUT GETTING 'CRUSHED'."
1700  PRINT "YOU, THE USER, WILL BE LOCATED "
1710  PRINT "AT THE CENTER OF THE VIDEO "
1720  PRINT "IN THE FORM OF A BLUE BLOCK.  "
1730  PRINT "YOUR ONLY MOVEMENTS WILL BE UP"
1740  PRINT "AND DOWN.  YOU WILL DO THIS "
1750  PRINT "WITH THE 'I' AND 'M' KEYS."
1770  PRINT
1780  PRINT "PRESS ENTER TO CONTINUE";
```

```
1790  INPUT X$
1800  HOME
1810  PRINT "TO TAKE ADVANTAGE OF ANY
      POINTS"
1820  PRINT "YOU MUST BE CONSTANTLY MOVING"
1830  PRINT "UP OR DOWN.  THE OUTER
      BOUNDARIES"
1840  PRINT "OF THE VIDEO WILL CONTAIN"
1850  PRINT "FLASHING LIGHTS FOR CONFUSION."
1860  PRINT "ALSO CONTAINED IN THE
      BOUNDARIES"
1870  PRINT "ARE BLOCKS, TWO OF WHICH"
1880  PRINT "WITHOUT NOTICE WILL MOVE VERY"
1890  PRINT "QUICKLY TOWARD THE CENTER.  TO"
1900  PRINT "YOUR LOCATION. IF YOU CAN "
1910  PRINT "MOVE OUT OF THE WAY, YOU CAN"
1920  PRINT "CONTINUE PLAYING; BUT IF, ON"
1925  FOR LX = 1 TO 2000: NEXT
1930  PRINT "THE OTHER HAND, YOU ARE
      CRUSHED,"
1940  PRINT "PLAY WILL ADVANCE TO THE NEXT"
1950  PRINT "PLAYER.  THIS IS ALSO TRUE IF"
1955  PRINT "YOU COLLIDE WITH THE BLOCKS."
1960  FOR LX = 1 TO 2000: NEXT LX
1970  PRINT "PRESS ENTER TO CONTINUE"
1980  INPUT X$
1990  HOME : PRINT "THE GAME WILL TERMINATE
      WHEN EACH"
2000  PRINT "PLAYER HAS PLAYEDA TOTAL OFTWO"
2010  PRINT "ROUNDS. THE PLAYER FINISHING "
2020  PRINT "WITH THE HIGHEST POINTS WILL BE"
2030  PRINT "THE WINNER..."
2040  PRINT
2050  PRINT "NOW ENTER THE FIRST NAMES"
2060  PRINT "OF THE ";PL;" PLAYERS:"
2070  FOR I = 1 TO PL: INPUT N$(I)
2080  NEXT
2090 PZ = 1
2100  PRINT N$(PZ);" WILL NOW PLAY. PRESS"
2105  IF PZ >  = 2 OR TR >  = 2 THEN 2150
2110  PRINT "THE ENTER KEY WHEN READY."
2120  INPUT S$
2130  HOME
2140  RETURN
2150  PRINT "THE ENTER KEY WHEN READY";
2160  INPUT X$:TI = 0: RETURN
2900  COLOR= 0: PLOT QU + I,F - 1: PLOT QW
      + 1,F - 1: PLOT QU + I,E - 1: PLOT QU
      + I,E: PLOT QU + I,F
```

```
 10  REM  PROGRAM TITLE:TRAPPED
 15  PRINT  CHR$ (4);"BLOADSOUND"
 20  HOME : CLEAR : DIM AA(12),PI(12),I$(12)
 30  PRINT "* TRAPPED *"
 40  PRINT : INPUT "LIST INSTRUCTIONS?";W$
 50  IF W$ = "Y" OR W$ = "YES" THEN  GOSUB
     3500
 60  HOME : GOSUB 3300:TU = 1
 70 QQ = 1:MM = 0:CR = 0
 80  GR
 90 CC = 0:BR = 0:TP = 1E + 04:TR = TP
100  REM  BORDER GRAPHICS
110 X = 1024:C = 204:V = 0
120  COLOR= 12: HLIN 0,39 AT 0
125  HLIN 0,39 AT 1
130  HLIN 0,39 AT 39
135  HLIN 0,39 AT 38
140  VLIN 0,39 AT 0
145  VLIN 2,37 AT 1
150  VLIN 0,39 AT 39: GOSUB 3450
155  VLIN 2,37 AT 38
160  REM  INNER GRAPHICS
165 C = 0
170 R = 2 * ( INT ( RND (1) * 13 + 4))
180 R1 =  INT ( RND (1) * 20 + 1)
185  IF C ) 0 THEN  GOTO 260
190 R2 = R1 + 9:R3 =  INT ( RND (1)
    * 9 + 1)
200 R4 = R1 + R3:R5 = R2 + R3
210 R6 =  INT ( RND (1) * 2 + 1)
220  IF R6 = 2 THEN R4 = R5
230  HLIN R1,R4 AT R
240 U = U + 1
250  IF U (  ) 5 THEN 170
260  HLIN 0,R1 AT R + 2
265  HLIN 0,R2 AT R + 4
270  HLIN R4,39 AT R - 2
275  HLIN R5,39 AT R - 4
277  IF C ) 2 THEN  GOTO 320
280  IF R )  = 12 THEN R = R - 4:C = C + 1:
     GOTO 180
285  IF R (  = 26 THEN R = R + 4:C = C + 1:
     GOTO 180
320  REM  START POSITION
330  GOSUB 3900
340 PC = ZZ: GOSUB 3900
345 PR = ZZ
```

```
350  IF  SCRN( PC,PR) <  > 0 OR  SCRN( (PC
     + 2),PR) <  > 0 THEN 330
355  IF  SCRN( PC,(PR + 1)) <  > 0 OR
     SCRN( PC,(PR - 1)) <  > 0 THEN 330
360 P1 = PC + 2:P2 = PR: COLOR= 9: PLOT PC,
    PR: COLOR= 13: PLOT P1,P2
370  REM  MOVEMENT OF COMPUTER PIECE
390  IF  SCRN( PC,(PR - 1)) = 0 THEN A =
     1: GOTO 440
400  IF  SCRN( PC,(PR + 1)) = 0 THEN A = 2:
     GOTO 440
410  IF  SCRN( (PC - 2),PR) = 0 THEN A =
     3: GOTO 440
420  IF  SCRN( (PC + 2),PR) = 0 THEN A =
     4: GOTO 440
430  GOTO 810
440  COLOR= 0: GOSUB 7000
441  PLOT PC,PR: IF  SCRN( P1,P2) = 0
     THEN 1700
442 XS =  PEEK ( - 16384) - 128: IF XS >
    0 THEN 900
445  GOTO 1130
450  POKE ( - 16368),0: ON A GOTO 460,
     470,480,490
460 PR = PR - 1: COLOR= 9: PLOT PC,PR:MM
    = 0: GOTO 710
470 PR = PR + 1: COLOR= 9: PLOT PC,PR:MM
    = 0: GOTO 400
480 PC = PC - 2: COLOR= 9: PLOT PC,PR:
    GOTO 760
490 PC = PC + 2: COLOR= 9: PLOT PC,PR:
    GOTO 420
710  REM  DOWN/UP-SEARCH LEFT/RIGHT
720  IF  SCRN( (PC - 2),PR) = 0 THEN 410
730  IF  SCRN( (PC + 2),PR) = 0 THEN 420
740  IF A = 1 THEN 390
750  GOTO 400
760  REM  LEFT/RIGHT-SEARCH DOWN/UP
770  IF  SCRN( PC,(PR - 1)) = 0 THEN 390
780  IF  SCRN( PC,(PR + 1)) = 0 THEN 400
790  IF A = 3 THEN 410
800  GOTO 420
810  REM  TRAPPED(COUNTER)
815 A(QQ) = A
820  IF A(QQ) = A(QQ - 1) THEN 880
830  IF MM >  = 8 THEN 1240
840 CR = CR + 10
860 QQ = QQ + 1: IF QQ >  = 3 THEN QQ = 1
```

```
 870  GOTO 390
 880  IF  SCRN( (PC + 2),PR) = 13 OR  SCRN
      ( (PC - 2),PR) = 13 THEN MM = MM + 1
 890  GOTO 830
 900  REM  USER PIECE/MOVE PER KEY
 910 KE =  PEEK ( - 16384) - 128
 920  IF KE = 77 THEN  POKE ( - 16368),0:
      GOTO 970
 930  IF KE = 73 THEN  POKE ( - 16368),0:
      GOTO 990
 940  IF KE = 75 THEN  POKE ( - 16368),0:
      GOTO 1010
 950  IF KE = 74 THEN  POKE ( - 16368),0:
      GOTO 1030
 960 CC = 0: GOTO 450
 970  IF  SCRN( P1,(P2 + 1)) (  ) 0 THEN 960
 980  GOSUB 7100: COLOR= 0: PLOT P1,P2:P2
      = P2 + 1: GOTO 1050
 990  IF  SCRN( P1,(P2 - 1)) (  ) 0 THEN 960
1000  GOSUB 7100: COLOR= 0: PLOT P1,P2:P2
      = P2 - 1: GOTO 1050
1010  IF  SCRN( (P1 + 2),P2) (  ) 0 THEN 960
1020  GOSUB 7100: COLOR= 0: PLOT P1,P2:P1
      = P1 + 2: GOTO 1050
1030  IF  SCRN( (P1 - 2),P2) (  ) 0 THEN 960
1040  GOSUB 7100: COLOR= 0: PLOT P1,P2:P1
      = P1 - 2
1050  COLOR= 13: PLOT P1,P2:TP = TP - 1
1070  REM  USERS MOVES (COUNTER)
1080 CC = CC + 1: IF CC )  = 10 THEN CC =
     0: GOTO 450
1090  GOTO 900
1130 CR = CR + 1
1140  IF CR )  = 1000 THEN  GOSUB 1450:
      GOTO 1180
1170  GOTO 450
1180  REM  RECHARGE COMPUTER PIECE
1190  COLOR= 11: PLOT PC,PR: FOR C = 1 TO
      200: NEXT
1200  GOSUB 7200
1210  COLOR= 0: PLOT PC,PR
1220  GOSUB 1510: GOSUB 3450: GOTO 450
1240  REM  TRAPPED?
1250  REM  CHECK CHARGE RATE FIRST
1260  IF CR )  = 925 THEN 1280
1270  GOTO 1290
1280 CR = 1000: GOTO 1370
1290  REM  TRAPPED COMPUTER PIECE
```

```
1300 L = 0
1310  COLOR= 0: PLOT PC,PR: FOR TI =
      1 TO 100: NEXT
1320  COLOR= 11: PLOT PC,PR:FOR TI = 1 TO 50
1340  NEXT
1350 L = L + 1: IF L <  = 2 THEN 1310
1360  GOTO 1600
1370  GOSUB 1450
1380 P = P + 6
1390  GOTO 450
1450  REM  MESSAGE
1460 MG = 1744:M$ = "RECHARGING"
1470  FOR I = 1 TO  LEN (M$)
1480  POKE MG, ASC ( MID$ (M$,I,1)) + 128
1490 MG = MG + 1: NEXT
1495  GOSUB 7300
1500  POKE MG,160: RETURN
1510 MG = 1744
1520  FOR I = 1 TO  LEN (M$)
1530  POKE MG,160
1540 MG = MG + 1: NEXT
1550  POKE MG,160: RETURN
1600  REM  MESSAGE (TRAPPED)
1610 MG = 1744
1620 M$ = "** TRAPPED **"
1630  GOSUB 1470
1640  GOTO 1910
1700  REM  USER "BURNED"
1710 MG = 1744
1720 M$ = "* USER BURNED *"
1730  GOSUB 1470
1740  REM  LEAVE USER POSITION AS-IS
1750 L = 0:MM = 0
1760  COLOR= 0: PLOT P1,P2: FOR TI = 1 TO
      100: NEXT
1770  COLOR= 13: PLOT P1,P2: FOR TI=1 TO 50
1790  NEXT : IF L <  = 1 THEN L = L + 1:
      GOTO 1760
1800 BR = BR + 1: GOSUB 1510: GOSUB 3450
1810  IF BR >  = 6 THEN  GOSUB 1830: GOTO
      1870
1820  GOTO 445
1830 L = 0
1840  REM  SOUND
1850  REM  SOUND
1860  HOME : RETURN
1870  PRINT
1880  PRINT "YOU'VE BEEN BURNED";BR;"TIMES";
```

```
1890  PRINT "SUCKER!!":TP = 0
1900  PRINT "YOU'RE OUT OF THE GAME...":
      GOTO 2000
1910  REM  TRAPPED
1920  GOSUB 1830
1930  PRINT "YA GOT ME TRAPPED, TURKEY!!!";
1940  PRINT "YOUR TOTAL POINTS NOW EQUAL ";
1950  PRINT TP
1960  PRINT "YOU MADE A TOTAL OF ";TR - TP;"
      MOVES"
1970  PRINT "BEFORE TRAPPING ME...."
2000  REM  PLAYER ADVANCE/POINTS
2010 PI(TU) = TP
2020  IF TU = N THEN 2100
2030 TU = TU + 1
2040  PRINT "PRESS RETURN, ";I$(TU)
2050  INPUT W$:CLS
2060  GOTO 70
2100  REM  ALL PLAYED
2110  REM  SORT NUMBERS/PLAYERS
2120  IF N = 1 THEN 2220
2130 F = 0:AA = 1
2140  IF PI(AA) >  = PI(AA + 1) THEN 2190
2150 P = PI(AA):PI(AA) = PI(AA + 1)
2160 PI(AA + 1) = P:I$ = I$(AA)
2170 I$(AA) = I$(AA + 1):I$(AA + 1) = I$
2180 F = 1
2190  IF AA < N THEN AA = AA + 1: GOTO 2140
2200  IF F = 1 THEN 2130
2210  GOTO 2260
2220  PRINT "YOU'VE PLAYED BY YOURSELF,
      AGAINST"
2230  PRINT "THE COMPUTER.  YOU WERE
      BURNED ";BR
2240  PRINT "TIMES.  FINAL TRAPPING SCORE:"
      ;PI(TU)
2250  GOTO 2330
2260 I = 1
2270  PRINT "BEST TRAPPER....";I$(I)
2280  PRINT "TOTAL POINTS=";PI(I)
2290  PRINT "FOLLOWING ";I$(I)
2300  FOR I = 2 TO N: PRINT I$(I);
2310  PRINT " WITH A SCORE OF:";
2320  PRINT PI(I)
2325  FOR TI = 1 TO 1000: NEXT TI,I
2330  PRINT
2340  PRINT "END OF PROGRAM RUN....."
2370  PRINT "UNLESS ALL WANT TO PLAY AGAIN???"
```

```
2380  PRINT "IF SO, TYPE RUN."
2390  PRINT : FOR TI = 1 TO 1500: NEXT
2400  PRINT "END"
2410  END
3300  REM  PLAYERS (NUMBER, NAMES)
3310  PRINT "ENTER NUMBER OF PLAYERS"
3320  INPUT N
3330  IF N (  = 0 OR N )  = 11 THEN 3310
3340  PRINT "NOW ENTER THE FIRST NAMES"
3350  PRINT "OF THE ";N;" PLAYERS"
3360  FOR I = 1 TO N
3370  INPUT I$(I): NEXT
3380  PRINT
3390  PRINT "THE NAME OF THE PLAYER WHO'S
      TURN"
3400  PRINT "IT IS WILL BE PRINTED AT THE
      BOTTOM"
3410  PRINT "OF THE GAME BOARD."
3420  FOR TI = 1 TO 1200: NEXT
3430  HOME
3440  RETURN
3450  REM  PLAYER
3460 MG = 1616
3470  FOR I = 1 TO  LEN (I$(TU))
3480  POKE MG, ASC ( MID$ (I$(TU),I,1)):MG
      = MG + 1
3490  NEXT : POKE MG,32: RETURN
3500  REM  INSTRUCTIONS
3510  PRINT "UP TO TEN PLAYERS CAN"
3520  PRINT "TRY TO OUTWIT THE COMPUTER AND"
3530  PRINT "MANEUVER ITS PIECE INTO A
      SITUATION"
3540  PRINT "WHERE IT WILL BECOME TRAPPED."
3550  PRINT "THE COMPUTER'S PIECE WILL LOOK "
3560  PRINT "LIKE AN ORANGE BLOCK.  YOU,
      THE USER,"
3570  PRINT "WILL HAVE A YELLOW BLOCK."
3580  PRINT "THE COMPUTER WILL MOVE ITS
      PIECE"
3590  PRINT "ALL AROUND THE GAME BOARD,
      CHANGING"
3600  PRINT "DIRECTION WHEN IT SENSES
      AN OBSTACLE."
3610  PRINT "YOU WILL FOLLOW THE COMPUTER'S
      PIECE"
3620  PRINT "AND TRY TO MANEUVER IT TO A
      SECTION"
```

```
3630  PRINT "OF THE GAME BOARD WHERE IT
      WILL BE"
3640  PRINT "TRAPPED--THAT IS, WHERE IT
      CAN ONLY"
3650  PRINT "MOVE LEFT OR RIGHT."
3660  PRINT
3670  INPUT "PRESS RETURN TO CONTINUE";W$:
      HOME
3680  PRINT "THE COMPUTER'S PIECE WILL
      CONTAIN A"
3690  PRINT "'CHARGE';IF YOU SIMULTANEOUSLY
      END AT"
3700  PRINT "THE SAME SPOT, YOU'LL GET '
      BURNED'."
3710  PRINT "SIX 'BURNS' AND YOU'RE OUT OF
      THE"
3720  PRINT "GAME. YOU WILL MOVE YOUR PIECE"
3730  PRINT "BY USING THE 'J' AND 'K' "
3740  PRINT "KEYS TO MOVE LEFT AND RIGHT,
      AND THE"
3750  PRINT "'I' AND 'M' KEYS TO MOVE UP
      AND DOWN."
3760  PRINT "WHENEVER YOU MOVE, THE
      COMPUTER WILL"
3770  PRINT "'BLANK' ITS PIECE.  YOU WILL
      BE ALLOWED"
3780  PRINT "ONLY TEN CONSECUTIVE MOVES
      AT ONE "
3790  PRINT "TIME.  WHEN YOU WISH TO MOVE,
      HOLD"
3800  PRINT "DOWN THE CORRECT MOVEMENT KEY."
3810  PRINT "POINTS WILL BE DETERMINED BY
      HOW"
3820  PRINT "MANY MOVES IT TAKES THE USER"
3830  PRINT "TO TRAP THE COMPUTER'S PIECE."
3840  PRINT "REMEMBER, TO TRAP THE
      COMPUTER'S PIECE,"
3850  PRINT "ONLY ALLOW IT TO MOVE LEFT
      AND RIGHT."
3860  INPUT "PRESS RETURN TO START:";W$
3870  HOME : RETURN
3900 ZZ =  INT (17 *  RND (1) + 20)
3910  RETURN
7000  POKE 769,25
7010  POKE 770,25
7020  POKE 768,1
7030  CALL 771
```

```
7040  RETURN
7100  POKE 769,93
7110  POKE 770,5
7120  POKE 768,1
7130  CALL 771
7140  RETURN
7200  FOR XQ = 130 TO 270 STEP 20
7205  IF XQ > 254 THEN XQ = 255
7210  POKE 769,XQ
7220  POKE 770,6
7230  POKE 768,10
7240  CALL 771
7250  NEXT
7260  RETURN
7300  FOR XQ = 1 TO 4
7310  POKE 769,20
7320  POKE 770,4
7330  POKE 768,5
7340  CALL 771
7350  POKE 769,56
7360  POKE 770,5
7365  POKE 768,5
7370  CALL 771
7440  NEXT
7450  RETURN

]
```

Elevator for the Apple II

```
 10  REM  PROGRAM TITLE: ELEVATOR
 20 P = 20: HOME
 30  DIM F(60),F$(60),G$(60)
 40  DIM KO(60)
 50  HTAB 10: PRINT "*** ELEVATOR ***"
 60  PRINT
 70  PRINT "INSTRUCTIONS REQUIRED (Y/N)";
 80  INPUT I$: IF I$ = "N" THEN N = 1
 90  INPUT "FIRST NAME: ";N$
100  PRINT : IF N = 1 THEN 230
110  INPUT "PRESENT OCCUPATION: ";P$
115  GOSUB 2150: HOME
120  PRINT "YOU HAVE JUST BEEN FIRED AS A ";
125  PRINT P$;"!!"
130  PRINT "BUT, YOU ARE IN LUCK. I HAVE JUST"
140  PRINT "HIRED YOU TO BE AN ELEVATOR"
145  PRINT "ATTENDANT IN MY COMPUTER-"
150  PRINT "IZED DEPARTMENT STORE. "
155  PRINT "YOUR JOB WILL BE SIMPLE,"
160  PRINT N$;".  ALL YOU HAVE TO DO "
170  PRINT "IS REMEMBER WHAT IS CON-"
180  PRINT "TAINED ON EACH OF SIXTEEN"
185  PRINT "FLOORS SO THAT YOU CAN TAKE"
190  PRINT "ALL OF MY CUSTOMERS TO THE"
200  PRINT "RIGHT LOCATION..."
210  PRINT
220  PRINT
230  REM  WHAT'S ON EACH FLOOR?
240  FOR I = 2 TO 19: READ F$(I): NEXT
250  IF N = 1 THEN 370
255  POKE ( - 16368),0: GOSUB 2150: HOME
260  PRINT "I ALMOST FORGOT TO TELL
     YOU--THE"
270  PRINT "CUSTOMERS, WHEN STEPPING ONTO
     THE"
280  PRINT "ELEVATOR, WILL TELL YOU WHAT
     THEY ARE"
290  PRINT "SHOPPING FOR.  THEY WON'T TELL
     YOU"
300  PRINT "AGAIN, BUT FOR EVERY CUSTOMER
     YOU PLACE"
310  PRINT "ON THE RIGHT FLOOR, YOU'LL
     BE PAID"
320  PRINT "$10.00.  THAT WILL BE YOUR ONLY
     WAGE!"
```

```
330  PRINT "CUSTOMERS WILL BE GETTING ON
     AND OFF"
340  PRINT "THE ELEVATOR AT DIFFERENT STOPS,
     AND"
350  PRINT "YOU WILL NEED TO REMEMBER
     THESE, TOO,"
355  PRINT N$;"!"
360  POKE ( - 16368),0: GOSUB 2150: HOME
370  PRINT "HERE ARE THE FLOORS AND
     CONTENTS:"
380  FOR I = 2 TO 16
390  PRINT I;") ";F$(I);
395  IF I = 8 THEN TT = I * 800: PRINT :
     GOSUB 2190
400  IF I >  = 2 AND I <  = 5 THEN 440
410  IF I >  = 6 AND I <  = 10 THEN  PRINT
     : GOTO 470
420  IF I >  = 11 AND I <  = 14 THEN 450
430  IF I >  = 15 AND I <  = 16 THEN 460
440  PRINT " ";F$(17): GOTO 470
450  PRINT " ";F$(18): GOTO 470
460  PRINT " ";F$(19)
470  IF RX = 1 THEN  RETURN
473  NEXT
475  REM  VARIABLE RX FOR PRINT ELEMENTS
480  FOR I = 1 TO 6: PRINT : NEXT :TT = TT
     + 2000
490  GOSUB 2190: HOME :C = 0:GK = 0
500  PRINT "YOU SHOULD BE ALL SET, ";N$
510 TT = TT - 1200: GOSUB 2190:RX = 1
520  REM  GET CUSTOMERS
530 C =  INT ( RND (1) * 10 + 1)
540  PRINT N$;", YOU NOW HAVE ";C;"
     CUSTOMER(S)"
550  PRINT "GETTING ONTO MY ELEVATOR."
560  REM  GET C RESPONSES
570 I = 1
580  REM  FLOOR NUMBER
590 F =  INT ( RND (1) * 14 + 2)
600 F(I) = F:I = I + 1
610  IF I <  = C THEN 590
620  REM  FLOOR TITLES
630  FOR I = 1 TO C
640 G$(I) = F$(F(I))
645 KO(F(I)) = KO(F(I)) + 1
650  NEXT
660  GOSUB 2190: PRINT
```

```
670  PRINT "THE CUSTOMER'S DESTINATIONS:"
     :FL = 1
680  PRINT : FOR I = FL TO C
690  PRINT G$(I);"";
700 F = F(I)
710  IF F )  = 2 AND F (  = 5 THEN 760
720  IF F )  = 11 AND F (  = 14 THEN 770
730  IF F )  = 15 AND F (  = 16 THEN 780
740  PRINT " FLOOR"
750  NEXT : GOTO 790
760  GOSUB 440: GOTO 750
770  GOSUB 450: GOTO 750
780  GOSUB 460: GOTO 750
790  GOSUB 2150
800  HOME
810  PRINT "PRESS (U) FOR (UP)"
820  PRINT "PRESS (D) FOR (DOWN)"
830  PRINT "THEN ENTER THE FLOOR NUMBER"
840 NN = 0: REM  DIRECTION
850  POKE ( - 16368),0: GOSUB 2160
860 D = A: POKE ( - 16368),0: IF D = 85
    THEN D$ = "UP": GOTO 870
865  IF D = 68 THEN D$ = "DOWN": GOTO 870
868  GOTO 950
870  PRINT "DIRECTION: ";D$
880  PRINT "FLOOR NUMBER";
890  INPUT FR
893  VTAB 12: HTAB 15:PRINT "PLEASE WAIT..."
895  IF FR = 1 AND TF = 0 THEN 1280
900  IF FR ( 1 OR FR ) 16 THEN 800
905  IF FR = TF THEN 1280
910  IF FR ( TF AND D$ = "UP" THEN 1260
915  IF FR ) TF AND D$ = "DOWN" THEN V = 1:
     GOTO 1260
920  GOSUB 2190: HOME : REM  PAUSE BEFORE
     START
930  IF D$ = "UP" THEN 970
940  IF D$ = "DOWN" THEN 1040
950  PRINT "SELECT A DIRECTION, ";N$;"!!"
960  PRINT : GOTO 810
970  REM  UP
980  IF TF )  = 17 THEN 1030
990  GOTO 1300
1000  PRINT "YOU ARE ON THE ";T$;:" FLOOR,
     ";N$
1010  PRINT "YOU CAN NOT GO ANY ";H$;"!!"
1020  GOTO 960
1030 T$ = "TOP":H$ = "HIGHER": GOTO 1000
```

```
1040  IF TF <  = 1 THEN 1060
1050  GOTO 1070
1060 T$ = "BOTTOM":H$ = "LOWER": GOTO 1000
1070  REM  DOWN CONTINUED MOVEMENT
1080 FS = P + 8:NN = 1: IF FS > 20 THEN FS
     = 20
1085  IF P >  = 20 THEN  GOTO 1190
1090  VTAB P: HTAB 20: PRINT FF
1100  FOR T = 1 TO 40: NEXT
1110  VTAB P: HTAB 20: PRINT "  ":P = P + 1
1120  IF P <  > FS THEN 1085
1130  REM  CONTINUE IF NOT ON FLOOR
1140  IF TF <  > FR THEN 1160
1150  GOTO 1500
1160 FS = FS + 7
1170  IF P >  = 20 THEN 1190
1180  GOTO 1090
1190  VTAB P: HTAB 20: PRINT " ":P = 4
1200 FS = P + 8:FF = FF - 1:TF = TF - 1
1210  GOTO 1090
1220  REM  NOT USED IN APPLE PROGRAM
1230  REM  NOT USED IN APPLE PROGRAM
1240  IF NN = 1 THEN 1100
1250  GOTO 1340
1260  PRINT "IF WANTING TO GO ";D$;", USE A"
1265  IF V = 1 THEN V = 0: PRINT "USE A
1270  PRINT "HIGHER NUMBER THAN THIS FLOOR"
1275  GOTO 1290
1280  PRINT "YOU ARE ON THAT FLOOR,";N$;"!!"
1290  GOSUB 2190: GOTO 800
1300  REM  UP MOVEMENT
1310  IF TF = 0 THEN 1460
1320 FS = P - 8: IF FS <  = 4 THEN FS = 4
1325  REM  NOT USED IN APPLE PROGRAM
1330  IF P <  = 4 THEN  GOTO 1430
1335  VTAB P: HTAB 20: PRINT FF
1340  FOR T = 1 TO 50: NEXT
1350  VTAB P: HTAB 20: PRINT "  ":P = P - 1
1360  IF P <  > FS THEN 1325
1370  REM  CONTINUE IF NOT ON FLOOR
1380  IF TF <  > FR THEN 1400
1390  GOTO 1500
1400 FS = FS - 7
1410  IF P <  = 4 THEN 1430
1420  GOTO 1325
1430  VTAB P: HTAB 20: PRINT "  ":P = 20
1440 FS = P - 8:FF = FF + 1:TF = TF + 1
1450  GOTO 1325
```

```
1460 TF = 1:P = 12:FF = 1
1470 FS = P - 7: GOTO 1325
1500  REM  ON FLOOR REQUESTED
1510  REM  NOT USED IN APPLE PROGRAM
1520  VTAB P: HTAB 20: PRINT FF: GOTO 1540
1530  VTAB P: HTAB 20: PRINT "1";
1540  FOR I = 1 TO C
1550  IF FR = F(I) THEN 1570
1560  NEXT :KO(FR) = KO(FR) - KG: GOTO 1600
1570 PA = PA + 10:KG = KG + 1
1580 F(I) = 0:XX = 1
1590  GOTO 1560
1600  REM  DID CUSTOMERS WANT OFF HERE?
1610  IF XX = 0 THEN 1630
1620 XX = 0: GOTO 1770
1630  PRINT N$;", NO CUSTOMERS WANTED TO
      GET OFF"
1640  PRINT "ON THIS FLOOR...":NC = NC + 1
1650  REM  MORE CUSTOMERS STEPPING ON
1660  IF FZ = 1 OR FR = 1 THEN AA = 1:FZ = 0
1670  IF AA = 1 THEN 1690
1680  GOTO 1730
1690 YU =  INT ( RND (1) * 5 + 1):YY = C:C =
     C + YU:C2 = YU
1700  IF C )  = 10 THEN HV = 1:C = C - C2:
      GOTO 1730
1710  REM  GET DESTINATIONS
1720  GOTO 1800
1730  REM  CONTINUE IF NOT FINISHED
1740 KG = 0: IF GK )  = C THEN 1940
1750  GOSUB 2190
1760  GOTO 800
1770 KG = KG + KO(FR):KO(FR) = 0
1775  PRINT KG;" CUSTOMERS JUST"
1780  PRINT "STEPPED OFF MY ELEVATOR, ";
1790  PRINT N$;"!": GOSUB 2700
1795 GK = GK + KG
1800  REM  YY MORE DESTINATIONS
1810 YU =  ABS (C - YY): IF GK )C THEN 1940
1815  IF YU (  = 0 THEN  PRINT "NO ONE GOT
      ON.": GOTO 2300
1820  PRINT N$;", YOU JUST HAD ";YU;" MORE"
1830  PRINT "CUSTOMERS STEP ONTO THE"
1840  PRINT "ELEVATOR.  THE DESTINATION(S):"
1850 KG = 0: PRINT : FOR I = YY + 1 TO C
1860 F =  INT ( RND (1) * 16 + 1)
1865  IF F = TF THEN 1860
1870  IF F = 1 THEN F$(1) = "GROUND":FZ = 1
```

```
1875 KO(F) = KO(F) + 1
1880 F(I) = F: NEXT : REM  FLOORS
1890  REM  TITLE OF FLOORS
1900  FOR I = YY + 1 TO C
1910 G$(I) = F$(F(I)): NEXT
1920 FL = YY + 1
1930  GOTO 680
1940  REM  STOP/CONT
1950  PRINT : PRINT "YOU HAVE NOW
      COLLECTED:"
1960  PRINT "$";PA: PRINT : GOSUB 2190
1965  PRINT : GOSUB 2500
1970  IF JV = 1 OR GK )  = C THEN 1990
1980  GOSUB 2190: GOTO 800: REM  ALL NOT OFF
1990  PRINT : GOTO 2300
2000  REM  DATA FOR FLOORS
2010  DATA  WOMENS,MENS,CHILDRENS,INFANTS
2020  DATA  TOYS, CARPETS,COMPUTERS
2030  DATA  APPLIANCES,FURNITURE,HARDWARE
2040  DATA  HOBBY,AUTO,OFFICE
2050  DATA  BATHROOM,BEDROOM
2060  DATA  CLOTHES,SUPPLIES,ITEMS
2070  REM  DATA ELEMENT 17 WILL BE USED
2080  REM  WITH ELEMENTS 2-5
2090  REM  DATA ELEMENT 18 WILL BE USED
2100  REM  WITH ELEMENTS 11-14
2110  REM  DATA ELEMENT 19 WILL BE USED
2120  REM  WITH ELEMENTS 15 & 16
2150  PRINT : PRINT "PRESS A KEY..."
2160 A =  PEEK ( - 16384) - 128
2170  IF A (  = 32 THEN 2160
2180  POKE ( - 16368),0: RETURN
2190  REM  TIME LOOP
2200  FOR T = 1 TO TT: NEXT
2210  GOTO 2180
2300  REM  CONTINUE?
2310  PRINT "WOULD YOU LIKE TO TAKE ON"
2320  PRINT "SOME MORE CUSTOMERS, ";N$;"?"
2330  PRINT "YOUR TOTAL EARNINGS SO FAR:";
2340  PRINT "$";PA;
2350  INPUT A$
2360  IF A$ (  ) "Y" AND A$ (  ) "YES" THEN
      2400
2370 TT = 2000:JV = 0: PRINT
2380  PRINT "YOU ARE ON FLOOR ";TF
2390  FOR I = 1 TO 16:KP = KP + KO(I):
      NEXT : IF KP ) 0 THEN  GOTO 1730
2395 NC = 0: GOTO 490
```

```
2400  PRINT
2410  PRINT "END OF ELEVATOR"
2420  END
2500  REM  POINT SCALE/PAY DEDUCTION
2510  IF NC (  = 0 THEN NC = 0: RETURN
2520  PRINT "YOU MADE";NC;" STOPS THAT WERE"
2530  PRINT "NOT NECESSARY, ";N$;". YOU"
2540  PRINT "WILL TAKE A DEDUCTION IN PAY"
2550  PRINT "OF $20.00 FOR EACH ONE OF"
2560  PRINT "THOSE STOPS, TOTAL";
2570 NN = NC * 20: PRINT "$";NN
2580  PRINT "LEAVING YOU ";
2590 PA = PA - NN
2595 NC = 0
2600  PRINT "$";PA: GOSUB 2190
2610  RETURN
2700  REM  ANY TIPS?
2710  IF KG (  = 1 THEN  RETURN
2720 AA =  INT ( RND (1) * 2 + 1)
2730  IF AA = 1 THEN  RETURN
2740 TI =  INT ( RND (1) * 100 + 1)
2750 PA = PA + TI
2760  PRINT
2770  PRINT "YOU'RE IN LUCK, ";N$;", A"
2780  PRINT "NICE CUSTOMER JUST GAVE "
2790  PRINT "YOU A TIP OF";
2800  PRINT "$";TI
2810  GOSUB 2190: RETURN
```

```
 10  REM  PROGRAM TITLE:FAREWELL
 20  HOME : HTAB 12: PRINT "*** FAREWELL ***"
 25  PRINT
 30  INPUT "NUMBER PLAYING (1-10)";A
 35  IF A < 1 OR A > 10 THEN 25
 40  PRINT "THE ";A;" PLAYERS' FIRST NAMES:"
 45  FOR I = 1 TO A
 50  INPUT Q$(I): NEXT
 55  PRINT :U = 1
 60  PRINT "ANY OF YOU NEED TO SEE"
 65  INPUT "THE INSTRUCTIONS? ";X$
 70  IF X$ <  > "Y" AND X$ <  > "YES" THEN
     210
 75  PRINT
 80  PRINT "READ CAREFULLY, IT COULD MEAN"
 85  PRINT "YOUR LIFE (OR THE GAME!!)..."
 90  PRINT "EACH PLAYER WILL SEE AND
     STUDY A"
 95  PRINT "MATRIX OF 6 BY 5 SQUARES
     (POSITIONS)"
100  PRINT "THESE WILL RANGE FROM THE "
105  PRINT "LETTERS A THROUGH Z WITH"
110  PRINT "FOUR SPACES BLANK.  TO PLAY:"
115  PRINT "THE COMPUTER WILL SELECT FIVE"
120  PRINT "POSITIONS AND PLACE COBRAS"
125  PRINT "WITHIN EACH. THE COMPUTER WILL"
130  PRINT "SHOW YOU WHERE THREE OF THE
     COBRAS ARE"
135  PRINT : INPUT "PRESS RETURN, PLEASE:";X$
140  HOME
145  PRINT "NOW THE COMPUTER WILL SELECT A"
150  PRINT "POSITION AND FLASH AN ASTERISK"
155  PRINT "THERE.  EACH PLAYER MUST ENTER
     FROM"
160  PRINT "MEMORY THE NUMBER WHICH WAS IN
     THAT"
165  PRINT "POSITION. IF CORRECT, THE PLAYER"
170  PRINT "GAINS 1000 POINTS; IF INCORRECT,
     THE"
175  PRINT "PLAYER LOSES 1000 POINTS. AND..
     .IF YOU"
180  PRINT "SELECT A POSITION WHERE
     THERE'S A"
185  PRINT "COBRA...YOU'RE OUT OF THE GAME,
     DEAD!"
```

```
190  PRINT "SELECTION WILL CONTINUE FOR AS
     LONG AS"
195  PRINT "EACH PLAYER WANTS OR UNTIL
     TWENTY-ONE"
205  PRINT "POSITIONS HAVE BEEN ELIMINATED."
210  INPUT "PRESS RETURN, PLEASE:";X$:
     GOSUB 220
215  HOME : GOTO 300
220  DIM L$(31),PR(31),PC(31),L(31)
225  REM  LETTERS?
228  IF SL <  > 0 THEN  PRINT "ARRANGING
     LETTERS..."
230 L = 65:I = 1:PZ = 1000
235 L$(I) =  CHR$ (L):L(I) = I
240  IF I = 27 THEN 255
245 I = I + 1:L = L + 1: GOTO 235
250  GOTO 235
255  REM  SORT LETTERS RANDOMLY
260 J = 1
265 N =  INT ( RND (1) * (I - 1) + 1)
270  IF L(N) = 0 THEN 265
275 L$(J) =  CHR$ (N + 64)
280 L(N) = 0:J = J + 1
285  IF J = I THEN  RETURN
290  GOTO 265
300  REM  POSITIONS
305 PC = 5:PR = 5:I = 1:PL = PR + 2:J =
     1:PK = PC
310 PR(I) = PR:PC(I) = PC:L(I) = I
315 PC = PC + 6:I = I + 1:J = J + 1
320  IF I = 27 THEN 345
325  IF J = 7 THEN 335
330  GOTO 310
335 PR = PL:PL = PL + 2:J = 1:PC = PK
340  GOTO 310
345  REM  THE REMAINING FOUR
350 L$(I) = " ":PR(I) = PR:PC(I) = PC:L(I) = I
355 PC = PC + 6:I = I + 1:J = J + 1
360  IF J = 7 THEN 370
365  GOTO 350
370  REM  MOVE THE FOUR BLANKS
375 I = 1:J = 1:L = 27
380 N =  INT ( RND (1) * 26 + 1)
385  IF L(N) = 0 THEN 380
390 F$ = L$(N):L$(N) = L$(L)
395 L$(L) = F$:L(N) = 0
400 L = L + 1: IF L = 31 THEN 410
405  GOTO 380
```

```
410  REM  NOW PRINT LETTERS
415  HOME : FOR I = 1 TO 30
420  VTAB PR(I): HTAB PC(I): PRINT L$(I):
     NEXT
425  REM  FINAL GRAPHICS
430 X = 2
435  FOR Y = 4 TO 13
440  VTAB Y: HTAB X: PRINT "@";: NEXT
445 X = X + 6
450  IF X )  = 40 THEN 460
455  GOTO 435
460  FOR X = 2 TO 38
465  VTAB Y: HTAB X: PRINT "@";: NEXT
470 Y = Y - 2
475  IF Y = 2 THEN 490
480  IF Y = 2 THEN 490
485  GOTO 460
490  REM  COBRA POSITIONS/PLAYER
495 C = 1: VTAB 18: HTAB 5: PRINT "PLAYER:
     ";Q$(U);
500 N =  INT ( RND (1) * (I - 5) + 1)
505  IF L(N) = 0 OR L(N) = 100 THEN 500
510 CB1(C) = PR(N):CB2(C) = PC(N):F$ =
     L$(N):C$(C) = F$
515 L$(N) =  CHR$ (42):L(N) = 100
520  IF C = 5 THEN 560
523  IF C (  = 3 THEN  GOSUB 530
525 C = C + 1: GOTO 500
530 XX = 0: REM  DEFINE 3 (1 FOR COMPUTER)
535  VTAB PR(N): HTAB PC(N): PRINT L$(N);:
     FOR T = 1 TO 100: NEXT
540  VTAB PR(N): HTAB PC(N): PRINT " ";:
     FOR T = 1 TO 50: NEXT
545  VTAB PR(N): HTAB PC(N): PRINT F$;
550  IF XX = 0 THEN XX = XX + 1: GOTO 535
555  RETURN
560  REM  COMPUTER SELECT/IF NOT DONE
565 SL = 1
575  IF SL )  = 22 THEN 1000
580  GOSUB 1200
585  VTAB 16: HTAB 5: PRINT "I WILL SELECT
     A POSITION:"
590 N =  INT ( RND (1) * (L - 1) + 1):
     GOSUB 1240
595  IF L(N) = 0 OR L(N) = 100 THEN 590
600 F$ = L$(N): IF SL = 1 THEN  GOSUB 530
605 C$ = F$:NU = N: GOSUB 1300: GOSUB 1200
```

```
610  VTAB 16: HTAB 5: PRINT "SELECT A
     POSITION BY LETTER ONLY, ";
615  VTAB 17: HTAB 5: PRINT Q$(U)
620  INPUT L$
622  GOSUB 1240: GOSUB 1200
625 I = 1
635  IF L$ = L$(I) THEN 665
645 I = I + 1
655  IF I >  = 31 THEN 675
660  GOTO 635
665 N = I:F$ = L$: GOSUB 530
670  GOTO 800
675  REM  COBRA POSITION OR BLANK
680  GOSUB 1200
685  GOSUB 1200
690  FOR I = 1 TO 5
695  IF L$ = C$(I) THEN 735
705  NEXT
715  VTAB 16: HTAB 5: PRINT "YOU CANNOT
     SELECT A BLANK!!"
725  GOSUB 1240: GOSUB 1200: GOTO 610
735  REM  COBRA
745  VTAB 16: HTAB 5: PRINT "YOU'RE DEAD,
     ";Q$(U);"!!!"
755  VTAB 17: HTAB 5: PRINT "THAT IS A
     COBRA POSITION!!!"
765  VTAB 18: HTAB 5: PRINT "** FAREWELL **"
     : GOSUB 1240
770 ZZ = 1: GOSUB 1200: GOTO 1020
800  GOSUB 1240: GOSUB 1200: IF L$ <  > C$
     THEN 900
805  VTAB 16: HTAB 5: PRINT "CONGRATULATIONS
     ";Q$(U);"..."
810  VTAB 17: HTAB 5: PRINT "YOU'VE JUST
     GAINED ";PZ;
815  PRINT "POINTS!!"
817  GOSUB 1240: GOSUB 1200
820  REM  PINTS/CONTINUE
825 W(U) = W(U) + PZ
830  VTAB PR(NU): HTAB PC(NU): PRINT " ";
835  VTAB 19: HTAB 20: PRINT "TOTAL POINTS:
     ";W(U)
840  GOSUB 1240: GOSUB 1200
845  VTAB 16: HTAB 5: PRINT "YOU NOW HAVE
     ";W(U);" POINTS..."
850  VTAB 17: HTAB 5: PRINT "...CONTINUE
     (Y/N)";
```

```
 855  INPUT X$: IF X$ = "N" THEN 1020
 860 SL = SL + 1:L(NU) = 0:L$(NU) = " "
 865  GOTO 575
 900  REM  WRONG POSITION
 905  VTAB 16: HTAB 5: PRINT "WRONG
      POSITION, ";Q$(U)
 910  VTAB 17: HTAB 5: PRINT "YOU ARE NOW
      MINUS ";PZ;
 915  PRINT " POINTS...": GOSUB 1240
 920  GOSUB 1200:W(U) = W(U) - PZ
 925  VTAB 19: HTAB 20: PRINT "TOTAL POINTS:
      ";W(U)
 930  GOTO 610
1000  REM  FINAL/ANOTHER PLAYER
1005  GOSUB 1240: GOSUB 1200
1010  VTAB 16: HTAB 5: PRINT "THAT'S IT,
      ";Q$(U);". YOU'VE"
1015  VTAB 17: HTAB 5: PRINT "REACHED THE
      LIMIT..."
1020  VTAB 18: HTAB 5: PRINT "PRESS RETURN,
      PLEASE:";
1025  INPUT X$: HOME
1030  IF ZZ = 1 THEN ZZ = 0: GOTO 1040
1035 T(U) = W(U)
1040 U = U + 1: IF A < U THEN 1100
1045  PRINT Q$(U);", YOU ARE NEXT..."
1050  PRINT "PRESS RETURN WHEN READY:";
1055  INPUT X$: HOME
1060  GOSUB 225: GOTO 300
1100  REM  FINAL/SCORE RESULTS
1105  PRINT "PLAYERS AND POINTS:"
1110  FOR I = 1 TO A
1115  PRINT Q$(I);" TOTAL POINTS: ";T(I)
1120  NEXT
1125  PRINT
1130  PRINT "PROGRAM TERMINATED..."
1135  END
1200  REM  CLEAR PRINT AREA
1210  VTAB 16: HTAB 5: FOR X = 1 TO 35:
      PRINT " ";: NEXT
1220  VTAB 17: HTAB 5: FOR X = 1 TO 35:
      PRINT " ";: NEXT
1230  RETURN
1240  REM  TIME DELAY
1250  FOR TI = 1 TO 1200: NEXT
1260  RETURN
1300  REM  FLASH SELECTION COURSE
1305  FOR I = 1 TO NU
```

```
1310  IF L$(I) = "*" OR L$(I)=" " THEN 1330
1315  VTAB PR(I): HTAB PC(I): PRINT "*";
1320  FOR TI = 1 TO 10: NEXT TI
1325  VTAB PR(I): HTAB PC(I): PRINT L$(I);
1330  NEXT I
1335  RETURN
```

```
 10  REM  PROGRAM TITLE:INSOMNIA
 20  CLEAR : HOME
 30  PRINT "INSOMNIA"
 40  PRINT
 50  PRINT "SO YOU ARE HAVING PROBLEMS"
 60  PRINT "SLEEPING?  JUST SIT BACK, KEEP"
 70  PRINT "YOUR EYES ON THE SCREEN AND"
 80  INPUT "PRESS RETURN:";X$: HOME
 90 A = 20:W = 0
100  PRINT : HTAB A: PRINT ":";
110  HTAB A - 1: PRINT ":";: HTAB A + 1:
     PRINT ":";
120 I =  INT ( RND (1) * 2 + 1)
130  IF I = 1 THEN 160
140  IF A )  = 39 THEN 170
150 A = A + 2: GOTO 180
160  IF A (  = 10 THEN 150
170 A = A - 2
180 W = W + 1: IF W = 15 THEN 350
190  IF W = 700 THEN 210
200  GOTO 100
210  IF W (  = 800 THEN 230
220  GOTO 250
230  PRINT
240 W = W + 1: GOTO 210
250  PRINT "GETTING SLEEPY NOW?"
260  FOR I = 1 TO W: NEXT
270  IF W (  = 814 THEN 290
280  GOTO 300
290  PRINT :W = W + 1: GOTO 270
300  FOR X = 1 TO 40
310  VTAB 1: HTAB X: PRINT "@": VTAB 23:
     HTAB X: PRINT "@": NEXT
320  FOR X = 1 TO 23
330  VTAB X: HTAB 1: PRINT "@": VTAB X:
     HTAB 40: PRINT "@": NEXT
340  FOR I = 1 TO W: NEXT :W = 0: GOTO 400
350  VTAB 16: HTAB 20
360  PRINT "KEEP EYES HERE!"
370  GOSUB 600
380  GOTO 190
400 L = 38:AR = 2:AC = 2:XR = 22:XC = 6:Y
     = AR
410  FOR I = AC TO AC + L
420  VTAB AR: HTAB I: PRINT "*";
```

```
430  VTAB XR: HTAB I + 4: PRINT "*"
440  NEXT : FOR I = AC TO AC + L
450  VTAB AR: HTAB I: PRINT " ";
460  VTAB XR: HTAB I + 4: PRINT " ";: NEXT
470 AR = AR + 1:XR = XR - 1: IF AR = 12
    THEN 530
480  IF XR = Y THEN 500
490  GOTO 410
500  IF W <  = 50 THEN 520
510  GOSUB 600: GOTO 610
520 W = W + 1: IF W = 25 THEN  GOSUB 580
525  GOTO 400
530  IF W > 1 THEN 480
540  VTAB AR: HTAB 10
550  PRINT "STARE AT THIS LOCATION";
560  GOSUB 600
570  GOTO 480
580 A2 = AR - 6: IF AR <  = 0 THEN AR = 1
585 A1 = AC + 25: IF A1 > = 40 THEN A1 = 40
590  PRINT "NOW FOLLOW THE BARS";
600  FOR I = 1 TO 1000: NEXT : RETURN
610  HOME
620  PRINT "IF YOU ARE NOT GETTING SLEEPY"
630  PRINT "BY NOW, YOU'RE A HOPELESS"
640  PRINT "CASE!"
650  GOSUB 600: PRINT
660  PRINT "THIS IS THE LAST PHASE OF"
670  PRINT "INSOMNIA.  IF IT FAILS, RUN"
680  PRINT "THE ENTIRE PROGRAM ONCE"
690  PRINT "AGAIN..."
700  GOSUB 600: PRINT
710  PRINT "KEEP YOUR EYES ON THE MOVING"
720  PRINT "BLOCK, THINKING TO YOURSELF"
730  PRINT "'I AM GETTING SLEEPY....'"
740  INPUT "PRESS RETURN:";X$:W = 0
750  HOME :PC = PC + 30
760 Y = 20: FOR X = 6 TO 9
770  VTAB X: HTAB Y: PRINT "@@": NEXT
780 PR = X:PC = 20
790  GOSUB 1100
800  FOR X = 6 TO 9
810  VTAB X: HTAB Y: PRINT "  ": NEXT
820  IF Z = 1 THEN 930
830  FOR X = 6 TO 9
840  VTAB X: HTAB Y: PRINT "@@":Y = Y - 1:
    NEXT
850  GOSUB 1120
860 PC = PC - 6
```

```
 870  GOSUB 1100
 880 Y = 20
 890  FOR X = 6 TO 9
 900  VTAB X: HTAB Y: PRINT "  ":Y = Y -
      1: NEXT
 910  GOSUB 1120
 920 Z = 1: GOTO 760
 930  GOSUB 1120
 940  FOR X = 6 TO 9
 950  VTAB X: HTAB Y: PRINT "@@":Y = Y + 1:
      NEXT
 960 PC = PC + 5: VTAB PR: HTAB PC: PRINT
      "@@"
 965  VTAB PR + 1: HTAB PC: PRINT "@@"
 970 Y = 20: FOR X = 6 TO 9
 980  VTAB X: HTAB Y: PRINT "  "
 990 Y = Y + 1: NEXT
1000  GOSUB 1120
1010 W = W + 1
1020  IF W = 5 THEN 1050
1030  IF W = 50 THEN 1150
1040 Z = 0: GOTO 760
1050  FOR X = 12 TO 27
1060  VTAB 6: HTAB X: PRINT "@": VTAB 12:
      HTAB X: PRINT "@": NEXT
1070  FOR X = 6 TO 11
1080  VTAB X: HTAB 12: PRINT "@": VTAB X:
      HTAB 27: PRINT "@": NEXT
1090  GOTO 1030
1100  VTAB PR: HTAB PC: PRINT "@@"
1105  VTAB PR + 1: HTAB PC: PRINT "@@"
1110  RETURN
1120  VTAB PR: HTAB PC: PRINT "  "
1125  VTAB PR + 1: HTAB PC: PRINT "  "
1130  RETURN
1150  REM  END OF PROGRAM
1155  VTAB 6: HTAB 20: PRINT "@@"
1160  GOTO 1160 REM  CONTINUOUS LOOP AT END
```

```
 10  REM  PROGRAM TITLE:RAINY DAZE
 20  HOME
 30  INPUT "INSTRUCTIONS? ";A$
 40  INPUT "HOW MANY PLAYERS? ";P: IF P = 0
     THEN 40
 50  PRINT "THE ";P;" PLAYERS' INITIALS: ";
 60  IF P >  = 11 THEN 20
 65  FOR I = 1 TO P
 70  INPUT N$(I): NEXT
 80  IF A$ <  > "Y" AND A$ <  > "YES"
     THEN 340
 90  REM  INSTRUCTIONS
100  PRINT "THIS PROGRAM IS DESIGNED TO"
110  PRINT "HELP YOU THROUGH YOUR WORST"
120  PRINT "DAYS, BY MAKING THEM WORSE!!"
130  PRINT "ACTUALLY, THIS PROGRAM WILL"
140  PRINT "TEST YOUR ABILITY TO FOLLOW"
150  PRINT "NUMBERS, SOMETIMES IN SEQUENCE."
160  PRINT "THE NUMERALS 1 THROUGH 660 WILL"
170  PRINT "RACE ACROSS AND FILL THE VIDEO."
180  PRINT "WHEN A NUMERAL APPEARS THAT IS"
190  PRINT "NOT SUPPOSED TO BE THERE, HOLD"
200  PRINT "THE RIGHT ARROW KEY FOR A SHORT"
210  PRINT "TIME UNTIL THAT NUMERAL IS"
220  PRINT "BLOCKED OUT."
230  PRINT
240  PRINT
250  INPUT "RETURN";X$
260  HOME
270  PRINT "POINTS WILL BE GIVEN ON
     THE BASIS"
280  PRINT "OF HOW MANY OUT-OF-SEQUENCE
     NUMERALS"
290  PRINT "YOU HAVE DETECTED AT THE END OF"
300  PRINT "THE COUNT, 660."
310  PRINT "REMEMBER TO KEEP GOOD EYE CONTACT"
320  PRINT "WITH THE SCREEN...";: PRINT
340 I = 1
350  IF P = 1 THEN 390
360  PRINT N$(I);", YOU WILL NOW PLAY:"
370  INPUT "PRESS RETURN, PLEASE:";X$
380  GOTO 410
390  PRINT "WHEN READY, ";N$(P);", PRESS"
400  INPUT "THE RETURN KEY:";X$
410  HOME :N = 0:N1 = 0:TT = 500
420 AR = 1:AC = 1:AB = AC:M = 30:L = 2:G = 0
```

```
430 K =  PEEK ( - 16384) - 128: POKE ( -
    16368),0: GOTO 550
440  VTAB AR: HTAB AC: PRINT N;
450  FOR T = 1 TO TT + TT
460  NEXT :AC = AC + 5
470  IF AC )  = M THEN 490
480  GOTO 430
490 AR = L:AC = AB:L = L + 1:TT = TT -
    15:G = 0
500  IF L )  = 24 THEN 520
510  GOTO 430
520  FOR T = 1 TO 1000: NEXT
530  HOME
540  GOTO 420
550  REM  CHANGE OR LEAVE
560  IF K = 21 THEN NJ = 1: GOTO 690
570 N = N + 1
580 N1 = N1 + 1: IF N1 (  = 2 THEN 610
590 J =  INT ( RND (1) * 2): IF G = 1 THEN
    J = 0:N = N1
600  IF J = 1 THEN N = N + 1:G = 1
610  IF K = 0 AND  ABS (N - N1) (  ) 0 THEN
    Y = Y + 1
620  IF NJ = 1 THEN NJ = 0: GOTO 640
630  IF N1 )  = 661 THEN 800
635  GOTO 440
640 N1 = N1 + 1
645  IF AC - 5 (  = 0 THEN  VTAB AR - 1:
    HTAB 40 - AC - 13: PRINT "@@@@@";: GOTO
650  VTAB AR: HTAB AC - 5: PRINT "@@@@@";
660  REM  PAUSE BEFORE CONT.
670  FOR T = 1 TO 200: NEXT :K = 0
680  GOTO 440
690  IF  ABS (N - N1) = 0 THEN 710
700 Y = Y - 1: GOTO 720
710 Y = Y + 1
720  GOSUB 970:A1 =  PEEK (P1 + (AC - 8))
730 A2 =  PEEK (P1 + (AC - 9))
740  IF A1 = 192 AND A2 = 192 THEN 760
750  GOTO 620
760  HOME :Y = Y + 2
770  PRINT "QUIT HOLDING DOWN THE ARROW"
780  PRINT "KEY, YOU MANIAC!!!"
790  FOR T = 1 TO 1200: NEXT : HOME : GOTO
    620
800  REM  PLAYER ADVANCE
810 YY(I) = Y:Y = 0: HOME
820 I = I + 1
```

```
 830  IF I > P THEN N1 = N1 - 1: GOTO 850
 840  GOTO 360
 850  REM  FINAL
 860  PRINT "PLAYERS AND THEIR POINTS"
 870  PRINT "ARE AS FOLLOWS:"
 880  FOR I = 1 TO P
 890  PRINT N$(I);" IDENTIFIED ";
 900  IF YY(I) = 0 THEN  PRINT "ALL
      NUMERALS. ": GOTO 930
 910  IF YY(I) < 0 THEN YY(I) =  - YY(I)
 920  PRINT N1 - YY(I);" NUMERALS."
 930  NEXT
 940  PRINT
 950  PRINT "END OF PROGRAM"
 960  END
 970  IF AR > 1 AND AR <  = 8 THEN P1 = AR
      * 128 + 896
 980  IF AR >  = 9 AND AR <  = 16 THEN P1
      = (AR - 8) * 128 + 936
 990  IF AR >  = 17 AND AR <  = 24 THEN P1
      = (AR - 16) * 128 + 976
1000  RETURN
```

```
 10  REM  PROGRAM TITLE: LAP THE TRACK
 20  CLEAR : HOME
 30  HTAB 10: PRINT "-:-: LAP-THE-TRACK
     :-:-"
 40  PRINT : DIM A$(15),A(15),LA(15),T(15)
 50  INPUT "INSTRUCTONS?";A$
 60  IF A$ <  > "Y" AND A$ <  > "YES" THEN
     300
 70  PRINT "UP TO FIFTEEN PEOPLE BET ON
     FOUR MOVING"
 80  PRINT "SPOTS THAT ARE IN A FAST RACE"
 90  PRINT "TO THE FINISH.  A COMPLETE RACE"
100  PRINT : IF CB = 1 THEN 960
110  PRINT "WILL BE ABLE TO CHOOSE A LANE
     (1-4)"
120  PRINT "AND MAKE A BET OF ANY DESIRED"
130  PRINT "AMOUNT ON THAT LANE.  IF THERE"
140  PRINT "ARE NO WINNERS, THE COMPUTER
     WILL"
150  PRINT "PLACE THE MONEY ON A LANE OF
     ITS CHOICE"
160  PRINT "IN THE NEXT RACE.  IF THE"
170  PRINT "COMPUTER'S SPOT HAPPENS TO WIN"
180  PRINT "IT WILL NOT TAKE THE MONEY;"
190  PRINT "IT WILL JUST BET ON ANOTHER
     LANE."
200  PRINT "SO, NO MATTER HOW YOU SLICE IT,"
210  PRINT "SOMEONE IS BOUND TO WIN.  (YOU
     CAN USE"
220  PRINT "REAL MONEY IN THIS GAME IF YOU
     WISH.)"
230  PRINT
300  PRINT
310  INPUT "HOW MANY WILL PLAY? ";B
320  IF B < 1 OR B > 10 THEN 320
330  PRINT "INITIALS, PLEASE: ";
340  FOR I = 1 TO B
350  INPUT A$(I): NEXT
360  PRINT
370  PRINT "YOU WILL NOW SELECT A LANE."
380 I = 1
390  PRINT A$(I);": LANE NUMBER";
400  INPUT LA(I)
410  IF LA(I) < 1 OR LA(I) > 4 THEN 390
420  PRINT "BET WHAT AMOUNT?";
430  INPUT B(I)
```

```
440 I = I + 1
450  IF I ) B THEN 480
460  PRINT
470  GOTO 390
480  HOME : GOSUB 1300: IF ZZ = 1 THEN 510
490  PRINT "IF MORE THAN ONE PLAYER WINS"
495  PRINT "(SELECTION OF SAME LANE), ALL
     MONEY"
500  PRINT "WILL BE DIVIDED AS EQUALLY AS
     POSSIBLE."
510  PRINT : IF CB = 1 THEN 960
520  GOSUB 1240: GOSUB 1300: GOSUB 1410
530  FOR TU = 1 TO 2000: NEXT : HOME
540  PRINT "PRESS RETURN TO START THE RACE:"
550  INPUT A$: HOME : GR
560  REM  OUTER TRACK
570  COLOR= 8: FOR I = 0 TO 30 STEP 10
580  HLIN 0,39 AT I
590  HLIN 0,39 AT I + 6
600  VLIN I,I + 6 AT 0
610  VLIN I,I + 6 AT 39
620  NEXT
630  REM  SPOT LOCATIONS
640  GOSUB 1200:L = 1:IR = 4:IC = 1
650 LR(L) = IR:LC(L) = IC:RR(L) = IR:RC(L)
    = IC:PR(L) = IR - 5:PC(L) = IC + 30:L
    P(L) = 0
655 PR(1) = 21:PR(2) = PR(1):PR(3) =
    22:PR(4) = PR(3):PC(1) = 16:PC(2) =
    36:PC(3 ) = PC(1):PC(4) = PC(2)
660 IR = IR + 10: IF L )  = 4 THEN 680
670 L = L + 1: GOTO 650
680  FOR I = 1 TO 4
690  COLOR= 3 * I: PLOT LC(I),LR(I): NEXT
700  REM  START A RACE
710 LP = 0:PK = LR(1) + 1
720 I =  INT ( RND (1) * 4 + 1):M =  ABS
    (I - 5)
730  COLOR= 0: IF LC(I) )  = 34 THEN  GOTO
     1000
735  PLOT LC(I),LR(I)
740  IF  ABS (LC(I) - RC(I)) )  = 29 THEN
     1000
750 LC(I) = LC(I) +  INT ( RND (1) * 10 + 1)
760  COLOR= 3 * I: IF LC(I) )  = 34 THEN
     GOTO 1000
765  PLOT LC(I),LR(I)
```

```
770  IF M + 1 < > I AND M + 1 < > 5 THEN
     790
780  GOTO 820
790  COLOR= 0: IF LC(M + 1) > = 34 THEN
     GOTO 1000
795  PLOT LC(M + 1),LR(M + 1)
800 LC(M + 1) = LC(M + 1) + INT ( RND
    (1) * 4 + 1)
810  COLOR= 3 * (M + 1): IF LC(M + 1) > =
     34 THEN GOTO 1000
815  GOSUB 2100: PLOT LC(M + 1),LR(M + 1)
820  IF M - 1 < > I AND M - 1 < > 0 THEN
840
830  GOTO 870
840  COLOR= 0: IF LC(M - 1) > = 34 THEN
     GOTO 1000
845  PLOT LC(M - 1),LR(M - 1)
850 LC(M - 1) = LC(M - 1) + INT ( RND
    (1) * 4 + 1)
860  COLOR= (M - 1) * 3: IF LC(M - 1) >
    = 34 THEN GOTO 1000
865  GOSUB 2160: PLOT LC(M - 1),LR(M - 1)
870  IF M + 2 < > 5 AND M + 2 < > 6 THEN
     890
880  GOTO 720
890  COLOR= 0: IF LC(M + 2) > = 34 THEN
     GOTO 1000
895  PLOT LC(M + 2),LR(M + 2)
900 LC(M + 2) = LC(M + 2) + INT ( RND (1)
    * 4 + 1)
910  COLOR= (M + 2) * 3: IF LC(M + 2) >
     = 35 THEN GOTO 1000
915  GOSUB 2220: PLOT LC(M + 2),LR(M + 2)
920  COLOR= 0: IF LC(M) > = 34 THEN GOTO
     1000
925  PLOT LC(M),LR(M)
930 LC(M) = LC(M) + INT ( RND (1) * 4 + 1)
940  COLOR= M * 3: IF LC(M) > = 34 THEN
     GOTO 1000
945  GOSUB 2280: PLOT LC(M),LR(M)
950  GOTO 720
960 CB = 0: TEXT : HOME : PRINT "THE
    COMPUTER WILL PLACE THE AMOUNT OF ";
965  PRINT "$";AM;
970 RF = INT ( RND (1) * 4 + 1)
980  PRINT "ON LANE ";RF;"."
985 MN(RF) = MN(RF) + AM:AM = 0
990  GOTO 510
```

```
1000  COLOR= 0: IF LC(I) >  = 34 THEN
      LC(I) = 38
1005  PLOT LC(I),LR(I)
1010 LC(I) = RC(I)
1020 LP(I) = LP(I) + 1: VTAB PR(I): HTAB
     PC(I): PRINT LP(I)
1030  IF LP(I) >  = 10 THEN 1500
1040  GOTO 720
1200  VTAB 21: HTAB 1: PRINT "LAPS - LANE
      1:";
1210  VTAB 21: HTAB 20: PRINT "LAPS -
      LANE 2:";
1220  VTAB 22: HTAB 1: PRINT "LAPS - LANE
      3:";
1230  VTAB 22: HTAB 20: PRINT "LAPS - LANE
      4:";
1235  RETURN
1240  REM  AMOUNTS BET PER LANE
1250  VTAB 21: HTAB 1: PRINT "LANE 1.....";
1260  VTAB 21: HTAB 20: PRINT "LANE 2.....";
1270  VTAB 22: HTAB 1: PRINT "LANE 3.....";
1280  VTAB 22: HTAB 20: PRINT "LANE 4.....";
1290  RETURN
1300  REM  AMOUNTS ON LANES
1310 T = 1
1320  FOR I = 1 TO 8
1330  IF LA(I) = T THEN 1350
1340  NEXT I: GOTO 1370
1350 MN(T) = MN(T) + B(I) / 2
1360  GOTO 1340
1370 T = T + 1
1380  IF T >  = 5 THEN 1400
1390  GOTO 1320
1400  RETURN
1410  REM  PRINT AMOUNTS
1420 AA$ = "$"
1430 I = 1:F1 = 21:F2 = 13
1440  VTAB F1: HTAB F2: PRINT AA$;MN(I)
1450 I = I + 1: IF I = 2 THEN F1 = 21:F2
     = 33
1453  IF I = 3 THEN F1 = 22:F2 = 13
1455  IF I = 4 THEN F1 = 22:F2 = 33
1460  IF I >  = 5 THEN  RETURN
1470  GOTO 1440
1500  REM  FINISH
1510 WW = 1
1515  FOR TU = 1 TO 1500: NEXT
```

```
1520  TEXT : HOME : VTAB 3: HTAB 15: PRINT
      "WINNER";
1530  FOR TU = 1 TO 1000
1540  NEXT
1550  VTAB 3: HTAB 15: PRINT "       ";
1560  IF WW )  = 6 THEN 1580
1570 WW = WW + 1: GOTO 1520
1580  REM  PLAYER LANE SELECT
1590 WW = 0
1600  FOR T = 1 TO 8
1610  IF LA(T) = I THEN 1630
1620  NEXT : GOTO 1650
1630 WW = WW + 1:T(WW) = T
1640  GOTO 1620
1650  FOR I = 1 TO 4:AM = AM + MN(I):MN(I)
      = 0
1660  NEXT : IF WW = 0 THEN 1760
1670  VTAB 3: HTAB 15: PRINT "WINNER(S):";
1680 W1 = 4:W2 = 15
1690  FOR I = 1 TO WW
1700  VTAB W1: HTAB W2: PRINT A$(T(I));
1710  IF WW ) 1 THEN  PRINT ",";
1720 W2 = W2 + 5: NEXT
1725  IF WW = 1 THEN 1790
1730  VTAB 17: HTAB 10: PRINT "EACH WINNER
      RECEIVES:";
1740  VTAB 17: HTAB 30: PRINT AA$;AM / WW;
1750 AM = 0: GOTO 1820
1760  VTAB 3: HTAB 15: PRINT "** NO WINNERS
      **          ";
1780  GOTO 1870
1790  VTAB 17: HTAB 10: PRINT ;:
1800  PRINT "YOU WILL RECEIVE:";
1810  GOTO 1740
1820  REM  ANOTHER
1830  VTAB 14: HTAB 15: INPUT "ANOTHER
      RACE (Y/N)?";A$
1850  IF A$ = "N" THEN 2000
1860  HOME :ZZ = 1: GOTO 360
1870  FOR TU = 1 TO 1200: NEXT
1880  VTAB 19: HTAB 1
1890  PRINT "IF ANOTHER RACE IS HELD, THE"
1910  PRINT "COMPUTER WILL BET THE MONEY"
1930  PRINT "ON A LANE IT SELECTS."
1940 CB = 1: GOTO 1820
2000  REM  END
2010  HOME
2020  PRINT "FAREWELL 'TILL NEXT RACE."
```

```
2030  END
2100  POKE 769,13
2110  POKE 770,30
2120  POKE 768,1
2130  CALL 771
2140  RETURN
2160  POKE 769,6
2170  POKE 770,8
2180  POKE 768,1
2190  CALL 771
2195  RETURN
2220  POKE 769,20
2230  POKE 770,4
2240  POKE 768,1
2250  CALL 771
2260  RETURN
2280  POKE 769,93
2290  POKE 770,5
2300  POKE 768,1
2310  CALL 771
2320  RETURN

  ]
```

```
 10  REM  PROGRAM TITLE:INJECTION
 20  CLEAR
 30  PRINT  CHR$ (4);"BLOADSOUND"
 40  HOME
 50  HTAB 15: PRINT ") INJECTION ("
 60  PRINT
 70  INPUT "NUMBER TO PLAY (1-10)";U
 80  IF U ( 1 OR U ) 10 THEN 70
 90  PRINT "THE INITIALS OF THE PLAYERS:"
100  FOR I = 1 TO U: INPUT K$(I)
110  NEXT : PRINT :PL = 1
120  PRINT "YOU ARE HEAD OF A LABORATORY"
130  PRINT "IN WHICH YOUR COMPANY IS IN THE"
140  PRINT "PROCESS OF INJECTING CUBES WITH"
150  PRINT "A SOLVENT, TO DESTROY A COLONY"
160  PRINT "OF BACTERIA WHICH CANNOT BE"
170  PRINT "CONTROLLED.  THESE CUBES WILL"
180  PRINT "RACE ACROSS THE SCREEN 5 AT A"
190  PRINT "TIME.  TO INJECT A CUBE, HOLD"
200  PRINT "THE SPACE BAR.THE INJECTED"
210  PRINT "SOLVENT MUST DIRECTLY STRIKE
     THE"
220  PRINT "OPENING AT THE BOTTOM OF THE
     CUBES."
230  PRINT
240  PRINT
250  INPUT "PRESS RETURN TO CONTINUE:";L$
270  PRINT "EACH TIME A CUBE IS INJECTED,"
280  PRINT "THE NUMBER RACING ACROSS THE
     SCREEN"
290  PRINT "WILL DECREASE BY ONE, BUT"
300  PRINT "NO FEWER THAN THREE WILL RACE"
310  PRINT "ACROSS AT ANY ONE TIME.  A
     TOTAL"
320  PRINT "OF TEN CUBES MUST BE INJECTED"
330  PRINT "WITH THE SOLVENT.  IF YOU MISS
     AN"
340  PRINT "INJECTION, THE NUMBER OF CUBES
     WILL"
350  PRINT "INCREASE BY ONE.  THE INJECTION"
360  PRINT "MACHINE CAN BE MOVED LEFT OR
     RIGHT"
370  PRINT "WITH THE LEFT OR RIGHT ARROW
     KEYS."
380  PRINT "GOOD LUCK!"
390  PRINT
```

```
400  INPUT "PRESS RETURN TO CONTINUE:";L$
405  GR
410  HOME :JN = 0:T = 1:IN = 10:E = 0:CV =
     5: GOSUB 600
420  REM  THE CUBES
430 X = 3:M = X:J = 0:JJ = 0: COLOR= 11
440  COLOR= 11: HLIN X,(X + 4) AT 2
450  HLIN X,(X + 4) AT 3
455  HLIN X,(X + 4) AT 4
460  COLOR= 0: PLOT (X + 2),4
470 X = X + 6:J = J + 1
475  GOSUB 1600
480  IF JJ = 1 THEN 520
490  IF J )  = CV THEN 510
500  GOTO 440
510  COLOR= 0: REM  BLANK OUT
520  HLIN M,(M + 4) AT 2
530  HLIN M,(M + 4) AT 3
535  HLIN M,(M + 4) AT 4
540 M = M + 6
550  REM  RECYCLE
560  IF X )  = 33 THEN X = 3:JJ = 1: GOTO
     580
570  IF M )  = 33 THEN 430
580  GOTO 720
600  COLOR= 12: REM  BORDER/INJECTOR
610  FOR X = 0 TO 39: PLOT X,0
620  PLOT X,38: PLOT X,39: NEXT
630  FOR X = 1 TO 39
640  PLOT 0,X: PLOT 1,X
650  PLOT 38,X: PLOT 39,X: NEXT
660  COLOR= 14: HLIN P,39 AT 15
670 QC = 2:QR = 14:CA = 0
680  PLOT QC,QR
690  VTAB 21: HTAB 1: PRINT "CUBES
     REMAINING ";
700  VTAB 21: HTAB 23: PRINT "CUBES ADDED ";
710  VTAB 22: PRINT K$(PL);: RETURN
720  REM  INJECTION/INJECTOR MOVE
730  IF R = 32 THEN 790
740 R =  PEEK ( - 16384) - 128: POKE ( -
    16368),0
750  IF R = 32 THEN 790
760  IF R = 8 THEN 900
770  IF R = 21 THEN 960
780  GOTO 440
790  IF JN = 1 THEN 810
```

```
 800 QQ = QC:JN = 1:Q1 = QR
 810  IF QR <  = 4 THEN V = 1: GOTO 860
 820  COLOR= 0: PLOT QC,QR
 830 QR = QR - 11: COLOR= 14: PLOT QC,QR
 833  GOSUB 1700
 835  IF  SCRN( (QC + 1),QR) = 11 THEN 845
 840  GOTO 440
 845  IF  SCRN( (QC - 1),QR) = 11 THEN 1020
 850  GOTO 440
 860  COLOR= 0: GOSUB 1900
 865  PLOT QC,QR
 870 QC = QQ + E:QQ = 0:E = 0:QR = Q1
 880  COLOR= 14: PLOT QC,QR: IF V = 1 THEN
      GOSUB 1140
 890 R = 0:JN = 0: GOTO 440
 900  REM  TO LEFT
 910  IF QC <  = 3 THEN 440
 920  COLOR= 0: PLOT QC,QR
 930 QC = QC - T
 940  COLOR= 14: GOSUB 1800
 945  PLOT QC,QR
 950  GOTO 440
 960  REM  TO RIGHT
 970  IF QC >  = 35 THEN 440
 980  COLOR= 0: PLOT QC,QR
 990 QC = QC + T
1000  COLOR= 14: GOSUB 1800
1005  PLOT QC,QR
1010  GOTO 440
1020  IF  SCRN( QC,(QR - 1)) = 11 THEN 1030
1025  GOTO 440
1030  IF  SCRN( (QC - 2),QR) = 11 THEN 1045
1040  GOTO 440
1045  IF  SCRN( (QC + 2),QR) = 11 THEN 1060
1050  GOTO 440
1060  REM  INJECTED
1070 T = T + 1:CV = CV - 1
1080  IF T >  = 5 THEN T = 1
1090  IF CV <  = 2 THEN CV = 5
1100 IN = IN - 1: VTAB 21: HTAB 18: PRINT "
     ";: HTAB 18: PRINT IN;
1110  IF IN <  = 0 THEN 1250
1120 E =  INT (5 *  RND (1) + 1): IF E + QQ >
     = 37 THEN E =  -  INT (5 *  RND 1) + 1)
1130  GOTO 860
1140  REM  CUBE ADDED/INJECTION MISSED
1150 IN = IN + 1
```

```
1160 CA = CA + 1:V = 0
1165  IF CA = 25 THEN  GOTO 1250
1170 T = T - 1
1180  IF T <  = 0 THEN T = 1
1190  VTAB 21: HTAB 18: PRINT IN;
1200  VTAB 21: HTAB 36: PRINT CA;
1210  RETURN
1250  REM  PLAYER ADVANCE
1260 W(PL) = CA:PL = PL + 1
1270  IF PL > U THEN 1310
1280  VTAB 22: PRINT "STANDBY ";K$(PL);"...
      ..";
1290  FOR YY = 1 TO 1500: NEXT
1300 R = 0: GOTO 410
1310  FOR YY = 1 TO 900: NEXT : TEXT : HOME
1320  PRINT "ALL PLAYERS HAVE HAD THEIR"
1330  PRINT "TURNS AT THE INJECTION
      MACHINE."
1340  PRINT "SCORES ARE BASED ON THE NUMBER
      "
1350  PRINT "OF CUBES WHICH WERE ADDED
      DURING"
1360  PRINT "THE INJECTION PROCESS."
1370  PRINT : FOR I = 1 TO U
1380  PRINT "INJECTOR:";K$(I)
1390  PRINT "HAD ";W(I);" CUBES ADDED."
1400  PRINT "FINAL SCORE EVALUATION:";
1410 W = W(I)
1420  IF W >  = 0 AND W <  = 5 THEN 1470
1430  IF W >  = 5 AND W <  = 10 THEN 1480
1440  IF W >  = 10 AND W <  = 20 THEN 1490
1450  PRINT "TOO BAD TO MENTION!";
1460  GOTO 1500
1470  PRINT "EXCELLENT!": GOTO 1500
1480  PRINT "FAIR.": GOTO 1500
1490  PRINT "POOR.": GOTO 1500
1500  FOR YY = 1 TO 1200: NEXT YY: PRINT
1510  NEXT I
1520  PRINT
1530  PRINT "END OF INJECTION"
1540  END
1600  POKE 769,23
1610  POKE 770,28
1620  POKE 768,2
1630  CALL 771
1640  RETURN
1700  POKE 769,112
1710  POKE 770,6
```

```
1720  POKE 768,1
1730  CALL 771
1740  RETURN
1800  POKE 769,25
1810  POKE 770,25
1830  POKE 768,1
1840  CALL 771
1850  RETURN
1900  POKE 769,132
1910  POKE 770,6
1920  POKE 768,1
1930  CALL 771
1940  RETURN
```

Appendix C

Making the Programs Work on Your Own Computer

Both the IBM and TRS-80 computers use MicroSoft BASIC, which means that they use similar syntax. However, you will find some differences as you program. Here are the most common problems you'll encounter in this book as you enter the programs which didn't come with special versions for the IBM or Apple.

CHR$. If a program asks for a character with an ASCII code smaller than 32 and larger than 90, it's probably some kind of special character limited to your computer. The computer industry hasn't bothered to standardize these numbers. TRS-80 ASCII codes between 129 and 191 are special graphics characters, and they're used to make things look interesting. Just substitute something your computer has that looks interesting to *you*! Check your basic manual for ideas. (My favorite IBM character is CHR$(2), a smiley face . . .)

CLEAR. This command just gets rid of all the "garbage" your computer has left lying around in its memory from other programs you've run. Don't bother to put a number after it unless your own basic manual suggests it—at least, not for any programs without machine language routines or a lot of high resolution graphics.

CLS. Apple users, use HOME!

DELAYS. Well, what did you buy IBM for if not for speed? But you'll have to almost double any time delay routines, such as for-next loops, unless you're twice as quick on the draw as the average computer owner.

ELSE. This very useful function is available to IBM users, but not to Apple lovers. Whenever you see a line in one of the original programs with an else statement, if the program is to run on an Apple II, you must create another line immediately following it to hold anything that follows the word else. Then you must make the original line skip that new line so that it makes it to where it was going in the first place. Did I lose you back at the pass? It works like this:

```
Original lines: 20 if A=10 THEN A=17 ELSE A=A+1:GOTO 10
                30 Z=INT(RND(1)*A+1)
New lines:      20 IF A=10 THEN A=17:GOTO 30
                25 A=A+1:GOTO 10
                30 Z=INT(RND(1)*A+1)
```

I hope you understood that, because that's all the explanation you are going to get.

INKEY$. This is another command Apple users must do without. It lets you monitor the keyboard directly without having to press the enter key. However, Apple users can peek and poke their way to a similar result. Instead of, for example, X$=INKEY$, you'll need to use X=PEEK(-16384)-128. X will be the ASCII code of whatever was input. Be sure to reset this memory location, though, or your program will think you've been holding down the REPT key. You do this by following your peek statement by POKE - 16368, 0.

PRINT@. See SCREEN FORMATTING

PRINT USING. This is another of those useful commands your Apple doesn't understand. This command allows you to format a number so that something going in looking like 987654321 can come out looking like $9,876,543.21. The print using statement will have a string of characters such as $$#####,.##, which works the conversion from chaos to order. Apple users will have to figure out what the number is supposed to look like from context and use regular print statements to approximate the TRS-80 results.

RND. TRS-80 BASIC accepts the statement RND(n) to produce a random number between n and 0. If you want the bottom value to be higher than zero, you must add a second statement such as "IF W<=10 THEN..." and go back to the RND statement until you get the value you want. A randomize statement in the early lines of the program will reset the random number generator so that it gives different values each time the program is run, otherwise the game may be too predictable.

Some BASICS, such as those used by IBM and Apple, can use a formula to produce random numbers within desired limits all in one step. Using the variable name A to represent the higher number and B for the lower, use this formula: X= INT(RND*(A–B)+B. Suppose you need random numbers between ten and twenty. 20–10=10, so X= INT(RND*10+10).

To ensure that the random number generator gives a different value each time the game is started, add the subscript (1) to RND on the Apple: X= INT(RND(1)*10+10). The IBM will require you to use the command Randomize n, where n is a number that can be input from the keyboard or be given a value by the program. See how it's done in the special IBM and Apple versions if you aren't sure how to write it into a program.

STRING$. I won't even say it . . . Do I hear the patter of little feet running out to trade in that Apple? No need! Think of all the terrific software you can get for it, and use a for-next loop.

STRING$ allows the computer to use one print statement to do a repetitive task, such as making a row of stars along the top and bottom of the screen. If the statement says PRINT STRING$ (40,42), it's telling the computer to print a line of forty characters with the ASCII code of 42, which is an asterisk. You can replace that statement with something like: FOR X=1 TO 40:PRINT "*";;NEXT. Don't forget the semicolon after the "*", or they'll all come out on different lines. (You already knew that, didn't you?)

SCREEN FORMATTING

The TRS-80 Model III Computer for which the programs in this book were originally written supports a screen display sixty-four characters wide and sixteen rows deep. The IBM, by contrast, offers a choice of either forty or eighty columns in width, with 24 rows. This means that programs which don't require any special formatting (i.e., graphics of some kind) can be used on the 80 width screen without changes. They may look a little lopsided, but that doesn't bother most people.

Most Apple computers have only a forty character screen. If you have an Apple or other narrow screen computer (or choose to use the 40 width screen on an IBM) or are entering a program which requires that the text be located on the screen in a precise manner, you will need to reformat.

First, decide what the program "wants". If the sample printout is all text, it probably doesn't matter how wide the lines are. If

they're too wide, they'll wrap around. If you fill the screen too soon and part of the text scrolls off, insert or move a pause routine (a for-next loop) or stop it with an input statement.

You have more screen lines, so you may want to center the text or graphics image by adding a few blank lines at the top ; or you can block text at the bottom in rows 17-23 (don't use line 24 unless you want a scrolling effect) to simplify a graphics image in the first sixteen rows.

Most of the complicated reformatting has been done for you in the separate program listings in Appendices A and B ; however, if you aren't sure how to format the remaining programs, use the following guides.

The most common TRS-80 screen formatting statement is PRINT @ . PRINT@locations start at the top left corner of the screen with PRINT@0, and work their way up one by one as they move across the screen to the right. The rightmost corner of the top line is PRINT@ 63. after which the count continues on the next row, starting again at the left. If you can't visualize this, look at the chart of PRINT@locations in Appendix D.

A screen location directly above or below another can be pinpointed by subtracting 64 from the original PRINT@location or adding 64 to it. If you want to convert the number into row and column figures you can use on the Apple (VTAB and HTAB locations) and the IBM (LOCATE statements), divide the PRINT @ value by 64 and add one to the answer to obtain a row, and add one to the remainder to get the column. Or look at the chart—after all, you paid for it.

TRS-80 also supports TAB(n) statements which are just like IBM's, and are the equivalent of Apple HTAB. However, whether you are looking at a PRINT@ location or a tab statement, if you are using a 40 character screen you will need to make sure your column values stay on the screen. Remember that 40 is approximately two-thirds of 64 ; reducing the column values by one-third and experimenting will probably get the desired result. If the program line wraps around and you don't want it to, either shorten it, or move the starting column position to the left.

TRS-80 also has a function called set, which lights up one pixel, or little light dot, on the screen. The locations are described in the same way Apple users describe plot locations, or IBM owners access PSET : SET X,Y means go over X and down Y and light up the screen there. Reset does just the opposite : it turns off a lighted pixel at the specified location.

However, since set and reset address pixels and plot addresses a block, you'll need to cut all X axis values (counting across the screen) by 2/3, and the Y values (counting down the screen) in half. You'll want to experiment a little. If the program says SET 127,30 followed by SET 100,30, the first dot will be at the right-hand edge of the screen, eleven rows down; and the second, while at the same level, will be about a fourth of the way toward the left side of the screen. The PRINT@ chart includes set information.

IBM users also have the option of converting set locations into locate statements, but they must remember to reverse the order of the axes, since locate statements give the row first, and then the column.

SCREEN GRAPHICS

Now the Apple users can watch the IBM users squirm for a minute. If you have an IBM PC with a monochrome monitor, it probably doesn't have a color/graphics card. This means that two of the special versions for IBM, *Knights* and *Intercept*, won't work on your screen. You will need to combine the information you learned in the section on screen formatting with the adapted versions to turn the screen graphics commands into text screen commands your system can display. Use your BASIC manual for a reference.

Also, almost all of the IBM programs use color, even when in the text screen. (This is because there are so many colors on the text screen!) If you have a graphics card and a "green screen" (not as crazy as it might sound), change all of the color statements to read COLOR 7,0,0. This will give you bright characters on a dark background. Or, even simpler, just don't type in any color statements at all.

SOUND EFFECTS

You can use the Apple sound routine for any programs in the book, whether they are written with sound or not. Look at the special Apple versions and see how the routine is loaded and called by the programs; then try your own. Many BASICS support sound statements. You'll see some of these in the IBM programs, which use the special musical abilities IBM comes with. Use these examples and those given in your BASIC manual to see what you can come up with for the programs you think are too quiet.

VARIABLE LENGTH

The world of computers is a world of trade-offs. Apple users

can name their programs anything they like, including multiple word titles; IBM users are stuck with eight letters and a suffix. But IBM users can name their variables anything they like, as long as it doesn't have spaces. You can expand variable names to make the program more meaningful if you want. Apple users, better stick to those two-letter names the programs come with.

And now, happy hacking!

Appendix D

Screen Locations

TRS-80 SCREEN LOCATIONS

	LEFT EDGE	1/4 ACROSS	1/2 ACROSS	3/4 ACROSS	RIGHT EDGE
ROW 1	:PRINT@ 0 SET 0 , 0	PRINT@ 15 SET 31 , 0	PRINT@ 31 SET 63 , 0	PRINT@ 47 SET 95 , 0	PRINT@ 63 SET 127 , 0
ROW 2	:PRINT@ 64 SET 0 , 3	PRINT@ 79 SET 31 , 3	PRINT@ 95 SET 63 , 3	PRINT@ 111 SET 95 , 3	PRINT@ 127 SET 127 , 3
ROW 3	:PRINT@ 128 SET 0 , 6	PRINT@ 143 SET 31 , 6	PRINT@ 159 SET 63 , 6	PRINT@ 175 SET 95 , 6	PRINT@ 191 SET 127 , 6
ROW 4	:PRINT@ 192 SET 0 , 9	PRINT@ 207 SET 31 , 9	PRINT@ 223 SET 63 , 9	PRINT@ 239 SET 95 , 9	PRINT@ 255 SET 127 , 9
ROW 5	:PRINT@ 256 SET 0 , 12	PRINT@ 271 SET 31 , 12	PRINT@ 287 SET 63 , 12	PRINT@ 303 SET 95 , 12	PRINT@ 319 SET 127 , 12
ROW 6	:PRINT@ 320 SET 0 , 15	PRINT@ 335 SET 31 , 15	PRINT@ 351 SET 63 , 15	PRINT@ 367 SET 95 , 15	PRINT@ 383 SET 127 , 15
ROW 7	:PRINT@ 384 SET 0 , 18	PRINT@ 399 SET 31 , 18	PRINT@ 415 SET 63 , 18	PRINT@ 431 SET 95 , 18	PRINT@ 447 SET 127 , 18
ROW 8	:PRINT@ 448 SET 0 , 21	PRINT@ 463 SET 31 , 21	PRINT@ 479 SET 63 , 21	PRINT@ 495 SET 95 , 21	PRINT@ 511 SET 127 , 21

	LEFT EDGE	1/4 ACROSS	1/2 ACROSS	3/4 ACROSS	RIGHT EDGE
ROW 9 :	PRINT@ 512 SET 0 , 24	PRINT@ 527 SET 31 , 24	PRINT@ 543 SET 63 , 24	PRINT@ 559 SET 95 , 24	PRINT@ 575 SET 127 , 24
ROW 10 :	PRINT@ 576 SET 0 , 27	PRINT@ 591 SET 31 , 27	PRINT@ 607 SET 63 , 27	PRINT@ 623 SET 95 , 27	PRINT@ 6 SET 127 ,
ROW 11 :	PRINT@ 640 SET 0 , 30	PRINT@ 655 SET 31 , 30	PRINT@ 671 SET 63 , 30	PRINT@ 687 SET 95 , 30	PRINT@ 703 SET 127 , 30
ROW 12 :	PRINT@ 704 SET 0 , 33	PRINT@ 719 SET 31 , 33	PRINT@ 735 SET 63 , 33	PRINT@ 751 SET 95 , 33	PRINT@ 767 SET 127 , 33
ROW 13 :	PRINT@ 768 SET 0 , 36	PRINT@ 783 SET 31 , 36	PRINT@ 799 SET 63 , 36	PRINT@ 815 SET 95 , 36	PRINT@ 831 SET 127 , 36
ROW 14 :	PRINT@ 832 SET 0 , 39	PRINT@ 847 SET 31 , 39	PRINT@ 863 SET 63 , 39	PRINT@ 879 SET 95 , 39	PRINT@ 895 SET 127 , 39
ROW 15 :	PRINT@ 896 SET 0 , 42	PRINT@ 911 SET 31 , 42	PRINT@ 927 SET 63 , 42	PRINT@ 943 SET 95 , 42	PRINT@ 959 SET 127 , 42
ROW 16 :	PRINT@ 960 SET 0 , 45	PRINT@ 975 SET 31 , 45	PRINT@ 991 SET 63 , 45	PRINT@ 1007 SET 95 , 45	PRINT@ 1023 SET 127 , 45

APPLE II SCREEN LOCATIONS

	LEFT EDGE	1/4 ACROSS	1/2 ACROSS	3/4 ACROSS	RIGHT EDGE
ROW 1 :	HTAB 1 PLOT 0 , 1	HTAB 10 PLOT 9 , 1	HTAB 20 PLOT 19 , 1	HTAB 30 PLOT 29 , 1	HTAB 40 PLOT 39 , 1
ROW 2 :	HTAB 1 PLOT 0 , 2	HTAB 10 PLOT 9 , 2	HTAB 20 PLOT 19 , 2	HTAB 30 PLOT 29 , 2	HTAB 40 PLOT 39 , 2
ROW 3 :	HTAB 1 PLOT 0 , 3	HTAB 10 PLOT 9 , 3	HTAB 20 PLOT 19 , 3	HTAB 30 PLOT 29 , 3	HTAB 40 PLOT 39 , 3
ROW 4 :	HTAB 1 PLOT 0 , 4	HTAB 10 PLOT 9 , 4	HTAB 20 PLOT 19 , 4	HTAB 30 PLOT 29 , 4	HTAB 40 PLOT 39 , 4

	LEFT EDGE	1/4 ACROSS	1/2 ACROSS	3/4 ACROSS	RIGHT EDGE
ROW 5 :	HTAB 1 PLOT 0 , 5	HTAB 10 PLOT 9 , 5	HTAB 20 PLOT 19 , 5	HTAB 30 PLOT 29 , 5	HTAB 40 PLOT 39 , 5
ROW 6 :	HTAB 1 PLOT 0 , 6	HTAB 10 PLOT 9 , 6	HTAB 20 PLOT 19 , 6	HTAB 30 PLOT 29 , 6	HTAB 40 PLOT 39 , 6
ROW 7 :	HTAB 1 PLOT 0 , 7	HTAB 10 PLOT 9 , 7	HTAB 20 PLOT 19 , 7	HTAB 30 PLOT 29 , 7	HTAB 40 PLOT 39 , 7
ROW 8 :	HTAB 1 PLOT 0 , 8	HTAB 10 PLOT 9 , 8	HTAB 20 PLOT 19 , 8	HTAB 30 PLOT 29 , 8	HTAB 40 PLOT 39 , 8
ROW 9 :	HTAB 1 PLOT 0 , 9	HTAB 10 PLOT 9 , 9	HTAB 20 PLOT 19 , 9	HTAB 30 PLOT 29 , 9	HTAB 40 PLOT 39 , 9
ROW 10 :	HTAB 1 PLOT 0 , 10	HTAB 10 PLOT 9 , 10	HTAB 20 PLOT 19 , 10	HTAB 30 PLOT 29 , 10	HTAB 40 PLOT 39 , 10
ROW 11 :	HTAB 1 PLOT 0 , 11	HTAB 10 PLOT 9 , 11	HTAB 20 PLOT 19 , 11	HTAB 30 PLOT 29 , 11	HTAB 40 PLOT 39 , 11
ROW 12 :	HTAB 1 PLOT 0 , 12	HTAB 10 PLOT 9 , 12	HTAB 20 PLOT 19 , 12	HTAB 30 PLOT 29 , 12	HTAB 40 PLOT 39 , 12
ROW 13 :	HTAB 1 PLOT 0 , 13	HTAB 10 PLOT 9 , 13	HTAB 20 PLOT 19 , 13	HTAB 30 PLOT 29 , 13	HTAB 40 PLOT 39 , 13
ROW 14 :	HTAB 1 PLOT 0 , 14	HTAB 10 PLOT 9 , 14	HTAB 20 PLOT 19 , 14	HTAB 30 PLOT 29 , 14	HTAB 40 PLOT 39 , 14
ROW 15 :	HTAB 1 PLOT 0 , 15	HTAB 10 PLOT 9 , 15	HTAB 20 PLOT 19 , 15	HTAB 30 PLOT 29 , 15	HTAB 40 PLOT 39 , 15
ROW 16 :	HTAB 1 PLOT 0 , 16	HTAB 10 PLOT 9 , 16	HTAB 20 PLOT 19 , 16	HTAB 30 PLOT 29 , 16	HTAB 40 PLOT 39 , 16
ROW 17 :	HTAB 1 PLOT 0 , 17	HTAB 10 PLOT 9 , 17	HTAB 20 PLOT 19 , 17	HTAB 30 PLOT 29 , 17	HTAB 40 PLOT 39 , 17

	LEFT EDGE	1/4 ACROSS	1/2 ACROSS	3/4 ACROSS	RIGHT EDGE
ROW 18 :	HTAB 1 PLOT 0 , 18	HTAB 10 PLOT 9 , 18	HTAB 20 PLOT 19 , 18	HTAB 30 PLOT 29 , 18	HTAB 40 PLOT 39 , 18
ROW 19 :	HTAB 1 PLOT 0 , 19	HTAB 10 PLOT 9 , 19	HTAB 20 PLOT 19 , 19	HTAB 30 PLOT 29 , 19	HTAB 40 PLOT 39 , 19
ROW 20 :	HTAB 1 PLOT 0 , 20	HTAB 10 PLOT 9 , 20	HTAB 20 PLOT 19 , 20	HTAB 30 PLOT 29 , 20	HTAB 40 PLOT 39 , 20
ROW 21 :	HTAB 1 PLOT 0 , 21	HTAB 10 PLOT 9 , 21	HTAB 20 PLOT 19 , 21	HTAB 30 PLOT 29 , 21	HTAB 40 PLOT 39 , 21
ROW 22 :	HTAB 1 PLOT 0 , 22	HTAB 10 PLOT 9 , 22	HTAB 20 PLOT 19 , 22	HTAB 30 PLOT 29 , 22	HTAB 40 PLOT 39 , 22
ROW 23 :	HTAB 1 PLOT 0 , 23	HTAB 10 PLOT 9 , 23	HTAB 20 PLOT 19 , 23	HTAB 30 PLOT 29 , 23	HTAB 40 PLOT 39 , 23
ROW 24 :	HTAB 1 PLOT 0 , 24	HTAB 10 PLOT 9 , 24	HTAB 20 PLOT 19 , 24	HTAB 30 PLOT 29 , 24	HTAB 40 PLOT 39 , 24

IBM PC SCREEN LOCATIONS

	LEFT EDGE	1/4 ACROSS	1/2 ACROSS	3/4 ACROSS	RIGHT EDGE
ROW 1 :	COLUMN 1 PSET 0 , 0	COLUMN 20 PSET 79 , 0	COLUMN 40 PSET 159 , 0	COLUMN 60 PSET 239 , 0	COLUMN 80 PSET 319 , 0
ROW 2 :	COLUMN 1 PSET 0 , 8	COLUMN 20 PSET 79 , 8	COLUMN 40 PSET 159 , 8	COLUMN 60 PSET 239 , 8	COLUMN 80 PSET 319 , 8
ROW 3 :	COLUMN 1 PSET 0 , 16	COLUMN 20 PSET 79 , 16	COLUMN 40 PSET 159 , 16	COLUMN 60 PSET 239 , 16	COLUMN 80 PSET 319 , 16
ROW 4 :	COLUMN 1 PSET 0 , 24	COLUMN 20 PSET 79 , 24	COLUMN 40 PSET 159 , 24	COLUMN 60 PSET 239 , 24	COLUMN 80 PSET 319 , 24
ROW 5 :	COLUMN 1 PSET 0 , 32	COLUMN 20 PSET 79 , 32	COLUMN 40 PSET 159 , 32	COLUMN 60 PSET 239 , 32	COLUMN 80 PSET 319 , 32

```
      LEFT EDGE       1/4 ACROSS      1/2 ACROSS      3/4 ACROSS      RIGHT EDGE
-------------------------------------------------------------------------------

ROW 6 :COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 40  PSET 79 , 40  PSET 159 , 40  PSET 239 , 40  PSET 319 , 40
-------------------------------------------------------------------------------

ROW 7 :COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 48  PSET 79 , 48  PSET 159 , 48  PSET 239 , 48  PSET 319 , 48
-------------------------------------------------------------------------------

ROW 8 :COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 56  PSET 79 , 56  PSET 159 , 56  PSET 239 , 56  PSET 319 , 56
-------------------------------------------------------------------------------

ROW 9 :COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 64  PSET 79 , 64  PSET 159 , 64  PSET 239 , 64  PSET 319 , 64
-------------------------------------------------------------------------------

ROW 10 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 72  PSET 79 , 72  PSET 159 , 72  PSET 239 , 72  PSET 319 , 72
-------------------------------------------------------------------------------

ROW 11 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 80  PSET 79 , 80  PSET 159 , 80  PSET 239 , 80  PSET 319 , 80
-------------------------------------------------------------------------------

ROW 12 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 88  PSET 79 , 88  PSET 159 , 88  PSET 239 , 88  PSET 319 , 88
-------------------------------------------------------------------------------

ROW 13 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 96  PSET 79 , 96  PSET 159 , 96  PSET 239 , 96  PSET 319 , 96
-------------------------------------------------------------------------------

ROW 14 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 104 PSET 79 , 104 PSET 159 , 104 PSET 239 , 104 PSET 319 , 104
-------------------------------------------------------------------------------

ROW 15 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 112 PSET 79 , 112 PSET 159 , 112 PSET 239 , 112 PSET 319 , 112
-------------------------------------------------------------------------------

ROW 16 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 120 PSET 79 , 120 PSET 159 , 120 PSET 239 , 120 PSET 319 , 120
-------------------------------------------------------------------------------

ROW 17 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 128 PSET 79 , 128 PSET 159 , 128 PSET 239 , 128 PSET 319 , 128
-------------------------------------------------------------------------------

ROW 18 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 136 PSET 79 , 136 PSET 159 , 136 PSET 239 , 136 PSET 319 , 136
-------------------------------------------------------------------------------

ROW 19 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 144 PSET 79 , 144 PSET 159 , 144 PSET 239 , 144 PSET 319 , 144
-------------------------------------------------------------------------------

ROW 20 :
       COLUMN 1       COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 152 PSET 79 , 152 PSET 159 , 152 PSET 239 , 152 PSET 319 , 152
-------------------------------------------------------------------------------
```

```
      LEFT EDGE       1/4 ACROSS      1/2 ACROSS      3/4 ACROSS      RIGHT EDGE
-------------------------------------------------------------------------------

ROW 21 :
       COLUMN 1        COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 160 PSET 79 , 160 PSET 159 , 160 PSET 239 , 160 PSET 319 , 160
-------------------------------------------------------------------------------

ROW 22 :
       COLUMN 1        COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 168 PSET 79 , 168 PSET 159 , 168 PSET 239 , 168 PSET 319 , 168
-------------------------------------------------------------------------------

ROW 23 :
       COLUMN 1        COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 176 PSET 79 , 176 PSET 159 , 176 PSET 239 , 176 PSET 319 , 176
-------------------------------------------------------------------------------

ROW 24 :
       COLUMN 1        COLUMN 20   COLUMN 40     COLUMN 60     COLUMN 80
 PSET 0 , 184 PSET 79 , 184 PSET 159 , 184 PSET 239 , 184 PSET 319 , 184
-------------------------------------------------------------------------------
```

Appendix E
Setting Up Your Games Disk

If you have a disk drive, why not make a menu-driven disk with all of the games in this book, just like the software you buy? If your computer's BASIC has the chain command, or if you know how to address the disk drive from a program, you can make a professional looking games disk. The IBM and Apple versions of a menu program are included in this section. You may use these programs as a general guide for creating menu programs for other systems.

MAKING A DISK FOR THE IBM PC

To make a disk for the IBM PC, format the disk using the FORMAT /S command, so that the disk will be self-booting. Next, make an auto-exec file by doing the following: With your DOS disk in the disk drive, type the command EDLIN AUTOEXEC.BAT and press the enter key. If you have two drives and the formatted disk is in drive B, be sure to type EDLIN B:AUTOEXEC.BAT. If you have one drive, take out the DOS disk and put in your formatted disk now.

Press I and the enter key; when you see the number 1:, type BASICA MENU (or whatever you plan to call menu program) and press enter. Now type E and press the enter key. The auto-exec file will be saved on the disk in the drive you specified. (I'll bet that was easier than you thought!)

You will need to copy BASICA.COM onto the disk; use the command COPY BASICA.COM B:; if you have two drives, the file will automatically copy onto the formatted disk in the B drive. If you

have one drive, put the formatted disk into the drive when instructed by the monitor.

Now, copy your game programs onto the newly formatted disk and add the menu program listed below. You will need to change all end statements in the game programs to CHAIN "MENU",10 (if you have used a different file name for your menu and a different starting line number, substitute those for "MENU" and "10"). Also, if you renumbered any programs, remember that the chain statement in the menu program must refer to an existing line number; check your program to see what line number you want the computer to chain to.

Menu Program for the IBM PC

```
 10 WIDTH 80:COLOR 7,0,0:KEY OFF:CLS
 20 LOCATE 10,25:PRINT "ADULT COMPUTER GAMES IN
    BASIC"
 30 PRINT:PRINT:PRINT TAB(38) "BY"
 40 PRINT:PRINT:PRINT TAB(32) "DAVID W. CHANCE"
 50 PRINT:PRINT:PRINT:PRINT:PRINT TAB(25)"IBM PC
    conversions by Tan A. Summers"
 60 PRINT:PRINT:INPUT "Press Enter For Menu",AA$
100 CLS:PRINT "1)  HOLD TIME":PRINT:PRINT "2)
    ELEVATOR":PRINT:PRINT "3)  FAREWELL":PRINT
110 PRINT "4)  MIND INVASION":PRINT:PRINT "5)
    KNIGHTS":PRINT:PRINT "6)  INJECTION":PRINT
120 PRINT "7)  POINT A TO POINT B":PRINT:PRINT "8)
    INTERCEPT"
130 PRINT:PRINT:PRINT "9) EXIT PROGRAM"
140 PRINT
150 PRINT "Press Space Bar for More Games"
160 R$=INKEY$
170 IF R$="" THEN GOTO 160
180 IF R$=" "THEN GOTO 200
190 R=VAL(R$)
195 ON R GOTO 310,320,330,340,350,360,370,380,390
200 CLS:PRINT "1)  INSOMNIA":PRINT:PRINT "2)
    LAP-THE-TRACK":PRINT:PRINT "3) BE PREPARED":PRINT
210 PRINT "4)  THE COURSE":PRINT:PRINT "5)  TRAPPED"
    :PRINT:PRINT "6)  RAINY DAZE ":PRINT
220 PRINT "7)  GENERAL STORE":PRINT:PRINT "8)
    RAIN OF TERROR"
230 PRINT:PRINT:PRINT "9)  EXIT PROGRAM"
240 PRINT
250 PRINT "Press Space Bar for More Games"
260 K$=INKEY$
270 IF K$="" THEN GOTO 260
```

```
280 IF K$=" "THEN GOTO 200
290 K=VAL(K$)
295 ON K GOTO 410,420,430,440,450,460,470,480,490
300 DEF SEG=0:IF (PEEK(&H410) AND &H30)<>&H30 THEN
    DEF SEG:GOTO 100
310 CHAIN "HOLDTIME",3
320 CHAIN "ELEVATOR",5
330 CHAIN "FAREWELL",5
340 CHAIN "MIND_INV",25
350 DEF SEG=0:IF (PEEK(&H410) AND &H30)<>&H30 THEN
    DEF SEG:GOTO 359
351 CLS:LOCATE 3,1
352 PRINT "THIS PROGRAM REQUIRES THE COLOR/GRAPHICS
    ADAPTER.  A SYSTEM CHECK"
353 PRINT "SHOW THAT YOU'RE NOT USING THAT ADAPTER.
    PRESS THE SPACE BAR"
354 PRINT "TO CONTINUE":DEF SEG:GOTO 356
355 IF INKEY$<>"" THEN GOTO 355
356 CMD$=INKEY$
357 IF CMD$="" THEN GOTO 356
358 IF CMD$=" " THEN CLS:GOTO 100
359 CHAIN "KNIGHTS",5
360 DEF SEG=0:IF (PEEK(&H410) AND &H30)<>&H30 THEN
    DEF SEG:GOTO 369
361 CLS:LOCATE 3,1
362 PRINT "THIS PROGRAM REQUIRES THE COLOR/GRAPHICS
    ADAPTER.  A SYSTEM CHECK"
363 PRINT "SHOW THAT YOU'RE NOT USING THAT ADAPTER.
    PRESS THE SPACE BAR"
364 PRINT "TO CONTINUE":DEF SEG:GOTO 366
365 IF INKEY$<>"" THEN GOTO 365
366 CMD$=INKEY$
367 IF CMD$="" THEN GOTO 366
368 IF CMD$=" " THEN CLS:GOTO 100
369 CHAIN "INJECTOR",5
370 CHAIN "POINT",5
380 CHAIN "INTRCPT",3
390 GOTO 999
410 CHAIN "INSOMNIA",20
420 CHAIN "LAPTRACK",3
430 CHAIN "BEPREP",5
440 CHAIN "COURSE",20
450 CHAIN "TRAPPED",5
460 CHAIN "RAINYDAZ",5
470 CHAIN "STORE",3
480 CHAIN "RAIN",20
490 GOTO 999
999 END
```

MAKING A DISK FOR THE APPLE II

Apple II users don't have to worry about formatting. Just write the menu program below with a new disk in the drive, and when you are satisfied, type INIT MENU (or whatever you want to call it), and press the return key. The menu program will automatically start each time the system is booted with that disk in the drive.

However, you probably don't have the chain command to work with. Instead, at the end of each program put PRINT CHR$(4); "RUN MENU" in place of the end statement. The PRINT CHR$(4) statement tells the disk drive to reload your menu program.

If you have another kind of computer and aren't sure how to make a menu-driven disk, try asking someone at the store where you bought your computer. Chances are, there's an easy way to do it for your computer, too.

```
10  TEXT : HOME
20  PRINT "1) MIND-INVASION"
30  PRINT "2) RAIN"
40  PRINT "3) LAPTRACK"
50  PRINT "4) KNIGHTS"
60  PRINT "5) INSOMNIA"
70  PRINT "6) ELEVATOR"
80  PRINT "7) FAREWELL"
90  PRINT "8) STORE"
100  PRINT "9) RAINYDAZE"
110  PRINT "10) TRAPPED"
120  PRINT "11) INJECTION"
130  PRINT "12) HOLDTIME"
140  PRINT "13) BE PREPARED"
145 D$ =  CHR$ (4)
150  PRINT : PRINT : PRINT "ENTER
      THE NUMBER OF THE GAME YOU
     WANT": INPUT "TO PLAY ";A
155  IF A > 14 OR A < 1 THEN 10
160  ON A GOTO 170,180,190,200,21
     0,220,230,240,250,260,270,28
     0,290,300,310,320
170  PRINT D$;"RUN MIND-INVASION"

180  PRINT D$;"RUN RAIN"
190  PRINT D$;"RUN LAPTRACK"
200  PRINT D$;"RUN KNIGHTS"
210  PRINT D$;"RUN INSOMNIA"
220  PRINT D$;"RUN ELEVATOR"
230  PRINT D$;"RUN FAREWELL"
```

```
240  PRINT D$;"RUN STORE"
250  PRINT D$;"RUN RAINYDAZE"
260  PRINT D$;"RUN TRAPPED"
270  PRINT D$;"RUN INJECTION"
280  PRINT D$;"RUN HOLDTIME"
290  PRINT D$;"RUN BE PREPARED"
300  HOME : END
```

Index

Index

A

Action games, 13, 42, 156, 182, 190, 195, 212, 306, 314
Amount on Board game, 271
Apple II, adapting programs for, 469

B

Be prepared for the Apple II, 415
Be Prepared for the IBM PC, 377
Be Prepared game, 182
Betting games, 271, 279

C

Call routine, 9
Chance games, 205
CHR$, 469
Clear command, 469
CLS, 469
Code games, 20, 28, 35, 165
Columns, 471
Course for the IBM PC, 381
Course game, the, 13

D

Dangerous Affair, 165
Date Service game, 121
Delay routines, 469
Disk set up for the Apple II, 484
Disk set up for the IBM PC, 481
Driver's Test game, 42

E

Elevator for the Apple II, 439
Elevator for the IBM PC, 331
Elevator game, 234
Else, 470
Experiments with sound, 3

F

Farewell for the Apple II, 446
Farewell game 249
From Point A to Point B game, 296

G

Games disks, 481
General interest, games of, 59
General Store for the Apple II, 404
General Store for the IBM PC, 394
General Store game, 92
Graphics, 473
Guessing games, 50, 104, 288

H

Hold Time, 156
Hold Time for the Apple II, 410
Hold Time for the IBM PC, 326

I

IBM PC, adapting programs for, 469
Injection for the Apple II, 464
Injection for the IBM PC, 352

Injection game, 306
INKEY$, 470
Insomnia for the Apple II, 452
Insomnia for the IBM PC, 369
Insomnia program, 258
Intercept for the IBM PC, 362
Intercept game, 314

K

Key a Date game, 131
Knights for the Apple II, 421
Knights for the IBM PC, 347
Knights game, 195

L

Lap-the-Track for the Apple II, 458
Lap-the-Track for the IBM PC, 372
Lap-the-Track game, 279
Love One Another game, 111

M

Master Code, 35
Math-a-Code game, 28
Mind Invasion for the Apple II, 426
Mind Invasion for the IBM PC, 342
Mind Invasion game, 212
Monochrome monitor, 473

N

Numbers, random, 471

P

Partners game, 288
Party games, 61, 75, 104, 269
Point A to Point B for the iBM PC, 356
Print , 470
Print using, 470
Program listings for the Apple II, 403
Program listings for the IBM PC, 325
PSET, 472

R

Rain of Terror, 190
Rain of Terror for the Apple II, 419
Rain of Terror for the IBM PC, 400
Rainy Daze for the Apple II, 455
Rainy Daze for the IBM PC, 391
Randomize statement, 470
Respect Your Computer, 104
Rnd, 470

S

Screen formatting, 471
Screen locations, Apple II, 476
Screen locations, IBM PC, 478
Screen locations, TRS-80, 475
Set statement, 472
Skill and logic, games of, 11
Skill games, 92, 141, 172, 222, 234, 249, 263, 296
Sound effects, 473
Sound experiments (TRS-80), 3
Sound routine for the Apple II, 9-10
Sound routine for the TRS-80, 1-3
Squares game, 141
STRING$, 471

T

Tab statement, 472
Trapped for the Apple II, 431
Trapped for the IBM PC, 384
Trapped game, 222
Triangle game, 205
Troubles game, 75

U

USR statements, 9

V

Variables, length of, 473

W

What Is It game, 50
Where's the Necessary Room game, 61
Width of screen, 471
Words-Zit game, 20

33 Adult Computer Games in BASIC

If you are intrigued with the possibilities of the programs included in *33 Adult Computer Games in BASIC for the TRS-80™, Apple II/IIe®, and IBM PC®* (TAB BOOK No. 1627), you should definitely consider having the ready-to-run tape or disk containing the software applications. This software is guaranteed free of manufacturer's defects. (If you have any problems, return the tape or disk within 30 days, and we'll send you a new one.) Not only will you save the time and effort of typing the programs, the tape or disk eliminates the possibility of errors that can prevent the programs from functioning. Interested?

Available on tape for the TRS-80, Models I and III, 16K. Available on disk for the TRS-80, Models I and III, 32K; for the Apple II, II plus, and *//e*, 32K; and for the IBM PC, 32K. Each tape or disk costs $19.95 plus $1.00 each shipping and handling. Each disk costs $34.95 plus $1.00 shipping and handling.

I'm interested in ready-to-run software for *33 Adult Computer Games in BASIC for the TRS-80™, Apple II/IIe®, and IBM PC®*. Send me:

_____ tape for the TRS-80, Models I and III, 16K (6034S)
_____ disk for the TRS-80, Models I and II, 32K (6035S)
_____ disk for the Apple II, II plus, and *//e*, 32K (6235S)
_____ disk for the IBM PC, 32K (6615S)
_____ TAB BOOKS catalog
_____ Check/Money Order enclosed for $ _____ (Include $19.95 for each tape or $34.95 for each disk plus $1.00 shipping and handling).
_____ VISA _____ MasterCard
Account No. _____ Expires _____
Name _____
Address _____
City _____ State _____ Zip _____
Signature _____
Mail To: **TAB BOOKS INC.**
Blue Ridge Summit, PA 17214

(Pa. add 6% sales tax. Orders outside U.S. must be prepaid with international money orders in U.S. dollars.)

TAB 1627

33 Adult Computer Games in BASIC for the IBM PC®, Apple II®///e® and TRS-80™

by David W. Chance

Get set for a great new computer game-playing experience . . . 33 exciting action-packed games and puzzles designed especially for adult interests and abilities! Quick thinking and plenty of imagination are needed as you play word and math games, fast action graphic games, guessing games, games of chance, even party games for up to 10 players.

Here are target and action games, a fascinating party game called "Doctor Chips" that has the computer assuming the role of your psychologist, a computer dating game, wildly funny games that will liven up any party, and much more. You can test your driving skills, find games that take full advantage of your micro's sound and graphics potentials, unravel mysterious codes, see if you can keep up with the computer as you take a whirlwind grocery shopping tour, play games of chance (that you may, or may not win), and play games like "Insomnia", "Rainy Daze", "Elevator", "Dangerous Affair", and many others.

Plus, these carefully written and fully explained programs make it easy for computer users to learn new programming techniques and to use different programming approaches for solving the same problem.

So, if you're looking for games that are designed for adult entertainment, rather than the "zap the aliens" variety that most youngsters prefer . . . if you're interested in livening up your next party with some truly unique computer entertainment . . . or if you'd like to improve your programming expertise while playing computer games, then this is the book you've been waiting for!

David W. Chance is a computer programmer who has written many game programs for a variety of computer systems. He is the author of TAB's *33 Challenging Computer Games for the TRS-80™/Apple™/PET®*.

OTHER POPULAR TAB BOOKS OF INTEREST

25 Graphics Programs in MICROSOFT® BASIC (No. 1533 — $10.95 paper; $17.95 hard)

33 Games of Skill and Chance for the IBM PC® (No. 1526 — $12.95 paper; $18.95 hard)

25 Exciting Computer Games in BASIC for All Ages (No. 1427 — $12.95 paper; $21.95 hard)

Writing BASIC Adventure Programs for the TRS-80 (No. 1422 — $9.95 paper; $14.95 hard)

TAB TAB BOOKS Inc.
Blue Ridge Summit, Pa. 17214

Send for FREE TAB Catalog describing over 750 current titles in print.

FPT > $13.50

ISBN 0-8306-1627-6

PRICES HIGHER IN CANADA

1295-1183